



**Ricardo
Monteiro**

Construção de uma plataforma robótica educacional





**Ricardo
Monteiro**

Construção de uma plataforma robótica educacional

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Eletrónica e Telecomunicações, realizada sob a orientação científica do Professor Doutor José Luís Costa Pinto Azevedo, Professor Auxiliar do Departamento de Eletrónica Telecomunicações e Informática da Universidade de Aveiro e do Professor Doutor Artur José Carneiro Pereira, Professor Auxiliar do Departamento de Eletrónica Telecomunicações e Informática da Universidade de Aveiro

o júri / the jury

presidente / president

Professor Doutor Filipe Miguel Teixeira Pereira da Silva

Professor Auxiliar da Universidade de Aveiro

vogais / examiners committee

Professor Doutor José Luís Costa Pinto de Azevedo

Professor Auxiliar da Universidade de Aveiro

Professor Doutor António Paulo Gomes Mendes Moreira

Professor Associado da Universidade do Porto - Faculdade de Engenharia

**agradecimentos /
acknowledgements**

À minha família,
À minha namorada,
Aos meus amigos.

Resumo

A robótica é usada com objetivos didáticos nos mais variados níveis de ensino. Potencia uma aprendizagem com uma forte componente prática ao mesmo tempo que é uma atividade lúdica, fomentando o divertimento e o espírito de entreajuda entre os intervenientes. Como veículo de ensino, as mais variadas plataformas estão disponíveis no mercado e têm sido construídas ao longo do tempo.

Na Universidade de Aveiro, realiza-se, desde 1995, o concurso Micro-Rato com objetivos didáticos, o qual tem incitado ao desenvolvimento de variadas plataformas robóticas.

Neste documento é descrito o processo de conceção, projeto e construção de uma plataforma robótica didática, tomando como inspiração as várias que têm sido construídas para o concurso Micro-Rato e aquilo que existe no mercado. Esta plataforma pretende ser uma mais valia para a robótica educacional, tendo baixo custo e facilidade de construção e expansão.

Abstract

Robotics is used with educational purposes among different school levels. It's a practical way of learning and it brings amusement to the people involved, they tend to have fun and help each others in the process of learning with robotics. As a teaching way, different robotic platforms can be found in the market and have been built.

The University of Aveiro holds since 1995 the Micro-Rato contest with educational purposes. Different platforms have been built taking it as an inspiration.

In this document it's described the process of conceptualizing, projecting and building a robotic platform with educational purposes taking inspiration from the different platforms built for the Micro-Rato contest and the ones available in the market. It is intended to build a platform which can be turn into a valuable educational object, with a low cost, easy to assemble and expand.

Conteúdo

Conteúdo	i
Lista de Figuras	iii
Lista de Tabelas	v
Lista de Acrónimos	vii
1 Introdução	1
1.1 Enquadramento e motivação	2
1.2 Objetivos	2
1.3 Trabalho realizado	3
1.4 Organização da dissertação	3
2 Robótica na educação	5
2.1 Ensino pré-universitário	6
2.2 Ensino universitário e investigação	7
2.3 Concursos de robótica	8
2.3.1 Trinity College Fire Fighting	9
2.3.2 RoboCup	10
2.3.3 Mobot Races	12
2.3.4 MicroMouse	13
2.4 Robótica educacional na Universidade de Aveiro	13
2.4.1 Concurso Micro-Rato e Ciber-Rato	14
2.4.2 Outras iniciativas de robótica educacional	15
2.5 Plataformas robóticas	16
2.5.1 Primeira plataforma Micro-Rato	16
2.5.2 Robô Didático de Aveiro (RoDA)	18
2.5.3 Pololu 3 π	19
2.5.4 EVALBOT	21
2.5.5 Arduino Robot	21
2.6 Conclusão	22
3 Projeto e Construção da Plataforma	25
3.1 Conceptualização da plataforma	25
3.2 Funcionalidades da plataforma	26
3.2.1 Suporte Sensorial	27

3.2.2	Atuação	27
3.2.3	Comunicação	27
3.2.4	Suporte de Processamento	28
3.2.5	Expansibilidade	28
3.3	Arquitetura da plataforma	28
3.3.1	Suporte sensorial	28
3.3.2	Atuação	40
3.3.3	Comunicação	44
3.3.4	Suporte para processamento	47
3.3.5	Alimentação da plataforma	49
3.4	Expansibilidade	56
3.5	Realização Física	57
3.6	<i>Software</i> de suporte	57
4	Resultados	67
4.1	Caraterização da plataforma	67
4.2	<i>Software</i> para teste da plataforma	72
4.3	Comparação com plataformas semelhantes	73
5	Conclusões e trabalho futuro	77
	Bibliografia	79
	Anexo 1	83
	Anexo 2	93
	Anexo 3	97
	Anexo 4	101
	Anexo 5	111
	Anexo 6	113
	Anexo 7	117

Lista de Figuras

2.1	Interior de uma casa, simulado para a realização de uma prova do concurso robô bombeiro	9
2.2	As várias modalidades do RoboCup Júnior	12
2.3	Pórtico na pista das Mobot Races, fonte [1]	12
2.4	Labirinto do MicroMouse, fonte [2]	13
2.5	Exemplos de labirintos dos concursos Micro-Rato	14
2.6	Limites máximos dos robôs participantes, fonte [3]	15
2.7	Plataforma Micro-Rato, fonte [4]	17
2.8	Plataforma RoDA	19
2.9	Plataforma Pololu 3 π	20
2.10	Plataforma robótica EVALBOT [5]	21
2.11	Plataforma Arduino Robot	22
3.1	Conceito base da plataforma, à esquerda vista de topo, à direita de trás	26
3.2	Característica da saída e funcionamento dos sensores Sharp	29
3.3	Zona de detecção e várias disposições dos sensores Sharp GP2D12	30
3.4	Duas disposições em ângulos diferentes dos sensores de obstáculos	31
3.5	Disposição dos sensores de obstáculos no robô, assinalados a tracejado preto, vista de topo.	31
3.6	Circuito de suporte para o sensor QRE1113 e sua saída característica	32
3.7	Caraterização elétrica do comportamento do sensor QRE1113	33
3.8	Disposição dos sensores de superfície no robô, assinalados a preto, vista de baixo.	34
3.9	Sinal emitido pelo farol de infravermelhos usado na competição Micro-Rato.	34
3.10	Posição do encaixe para o suporte do sensor.	35
3.11	Estrutura necessária para o suporte do sensor.	35
3.12	Diagrama de blocos interno do sensor GP1UE28YK e circuito detetor de envolvente	35
3.13	Circuito simulado e gráfico resultante da simulação	36
3.14	<i>Encoder</i> usado, montado num motor e vista explodida	37
3.15	Diagrama temporal de uma operação de leitura do <i>encoder</i> . [6]	38
3.16	Valores lidos do <i>encoder</i> com leituras de 4 bits, valores em base hexadecimal	39
3.17	Posição dos botões na plataforma, assinalados a preto numa vista de topo.	40
3.18	Modelo de locomoção diferencial com dois <i>casters</i>	40
3.19	Funcionamento de uma Ponte-H	41
3.20	Esquema do compensador usado	42
3.21	Constituintes de um servo-motor. Fonte: [7]	43
3.22	Posição dos LEDs e <i>Buzzer</i> na plataforma Micro-Rato	44

3.23	Esquema do módulo de interface <i>ZigBee</i> com a entidade externa	45
3.24	Posição das diferentes partes da comunicação na plataforma. Vista de topo	47
3.25	Esquema da unidade de alimentação da plataforma	50
3.26	Esquema do circuito de monitorização do nível das baterias	51
3.27	Circuito de comutação da alimentação das tensões da eletrónica de baixa tensão	51
3.28	Circuito tipo para implementar um regulador <i>step-up</i> baseado no chip MAX669	53
3.29	Circuito para implementar um regulador <i>step-up</i> baseado no chip MAX669	54
3.30	Forma de onda da corrente na bobina <i>L3</i> , frequência de <i>switching</i> de 500 kHz	55
3.31	Posição dos diversos componentes na plataforma robótica	57
3.32	Estrutura global do software desenvolvido	58
3.33	Esquema do software que implementa o protocolo de comunicação na plataforma	62
3.34	Esquema do software que implementa o protocolo de comunicação na entidade externa	64
3.35	Esquema conceptual da <i>Application Programming Interface</i> (API) da entidade externa	65
4.1	Várias vistas do protótipo construído	68
4.2	Gráficos da velocidade da plataforma para diferentes <i>setpoints</i>	69
4.3	Gráficos obtidos com os dados dos sensores de obstáculos e seu processamento	71
5.1	Diagrama temporal de uma operação de leitura do <i>encoder</i> . [6]	111

Lista de Tabelas

3.1	Comparação entre várias alternativas para medição de distâncias. Fonte: [8]	29
3.2	Parâmetros do sistema de locomoção. Fonte: [9]	38
3.3	Resumo de parâmetros usados para obtenção de dados dos <i>encoders</i>	38
3.4	Características principais dos motores GHM-03. Fonte: [9]	41
3.5	Características principais do servo-motor HS-5055MG. Fonte: [10]	43
3.6	Comparação entre BlueTooth e ZigBee. Fontes: [11] e [12]	45
3.7	Sequências de comandos para configuração de um link permanente entre dois módulos <i>XBee</i>	46
3.8	Características mínimas da unidade de controlo	48
3.9	Características do microcontrolador PIC32MX795F512H	49
3.10	Consumo de corrente das várias partes da plataforma robótica	52
3.11	Consumo de corrente das várias partes da plataforma robótica	54
4.1	Resumo das características da plataforma decorrentes do seu projeto	70
4.2	Valores de m e b obtidos para os vários sensores de obstáculos	71
4.3	Potência consumida pela plataforma no ensaios realizados	72
4.4	Comparação da plataforma desenvolvida com as plataformas disponíveis no mercado	74
5.1	Tempos e frequências associadas ao <i>encoder</i> AEAT-6012. Fonte: [6]	111
5.2	Comandos para comunicação da plataforma para a entidade externa.	114
5.3	Campos de dados da trama de comunicação dos dados sensoriais	114
5.4	Comandos para comunicação da entidade externa para a plataforma.	115
5.5	Campos de dados da trama de comunicação dos dados sensoriais	116
5.6	Preço do componentes eletrónicos necessários para a montagem do robô	118

Lista de Acrónimos

UA	Universidade de Aveiro
CWRU	<i>Case Western Reserve University</i>
MIT	<i>Massachusetts Institute of Technology</i>
FNR	Festival Nacional de Robótica
DETI	Departamento de Eletrónica Telecomunicações e Informática
PCB	<i>Placa de Circuito Impresso</i>
CMU	<i>Carnegie Mellon University</i>
ADC	<i>Analog-to-Digital Converter</i>
RoDA	Robô Didático de Aveiro
ASK	<i>Amplitude Shift Keying</i>
RAM	<i>Random-access Memory</i>
SSI	<i>Synchronous Serial Interface</i>
SPI	<i>Serial Peripheral Interface</i>
PSD	<i>Position Sensitive Detector</i>
LED	Light Emitting Diode
PWM	Pulse Width Modulation
USB	<i>Universal Serial Bus</i>
UART	<i>Universal Asynchronous Receiver Transmitter</i>
ASCII	<i>American Standard Code for Information Interchange</i>
ADC	<i>Analog to Digital Converter</i>
I2C	<i>Inter-Integrated Circuit</i>
CAN	<i>Controller Area Network</i>
API	<i>Application Programming Interface</i>

ARM *Acorn RISC Machine*

UTAD Universidade de Trás-os-montes e Alto Douro

LCD *Liquid-crystal display*

Capítulo 1

Introdução

A robótica revolucionou o mundo pela forma como permitiu a automatização de muitas tarefas que de outra forma exigiriam um imenso esforço humano e também acarretariam uma série de riscos. O esforço para automatizar processos e tarefas vem de civilizações antigas quer em ideia quer praticamente. Para os gregos, o deus Hefesto criou uma série de serventes mecânicos, pois era manco e precisava de ajuda para se mover [13]. Não só na mitologia, foram os gregos os autores do mecanismo *Antikythera* capaz de prever posições de planetas no céu [14]. Mais uma vez no plano imaginário, na República Checa foi forjado o nome robô em 1920 pelas mãos de Karel Čapek na sua obra teatral “*Rossum’s Universal Robots*” em que eram criados serventes por meio da biotecnologia, estes eram obrigados a trabalhar pelos humanos [15].

Esta ideia dos robôs trabalharem pelas pessoas tem sido alvo de enorme discussão ao longo dos tempos. Há certos problemas e desafios quando se trata de equipar uma indústria com automatismos para efetuar o trabalho. Por um lado, estes automatismos podem estar a substituir trabalho humano, o que leva a desemprego e lucro maior para a indústria, por outro lado, pode ao mesmo tempo ser criado emprego e lucro pois os automatismos potenciam a abertura de novos negócios que antes não existiam. Veja-se o caso das fotocopiadoras: a introdução da máquina veio potenciar lucro ao dono da empresa ao mesmo tempo que criou postos de trabalho para os operadores. Esta discussão é feita em mais pormenor em [16] no primeiro capítulo onde são abordadas questões sociológicas e filosóficas respeitantes à robótica.

A educação tem também a ganhar com a introdução da robótica nas salas de aula. No livro já falado [16], o uso dos robôs nas salas de aulas restringia-se a três aplicações. A primeira era a linguagem de programação *Karel* uma modificação de Pascal que pretendia definir um ambiente robótico, onde um robô devia escapar de um local passando por robôs hostis e ajudado por três robôs amigos. A segunda era a linguagem *LOGO* em que se pretendia que um robô desenhasse polígonos, servindo principalmente para ensinar matemática de uma forma divertida, acabando por ensinar bases de programação. Finalmente a terceira tratava-se de uma série de manipuladores robóticos e robôs móveis.

Este livro foi escrito em 1991 e desde então o uso de robôs na educação tem conhecido um grande crescimento de aplicações e objetivos. Aqui enquadra-se o trabalho que foi desenvolvido nesta dissertação.

1.1 Enquadramento e motivação

Ao longo dos anos tem-se procedido a uma introdução dos robôs nas salas de aula. Usando estas máquinas, os alunos ganham vantagens educativas por lidarem com *hardware*, fazendo um trabalho de “mãos na massa”, permite-lhes de uma forma rápida detetarem problemas e chegarem a soluções para esses problemas, testando a sua validade logo de seguida. As capacidades de projeto dos alunos saem reforçadas pois são chamados a criar soluções inovadoras de uma forma motivante. Ao mesmo tempo, uma vez que muitos destes trabalhos são feitos em grupo, a capacidade de trabalhar em grupo sai muito reforçada. Por ser uma área bastante transversal envolvendo conhecimentos de matemática, eletrónica, mecatrónica, mecânica e programação, entre outros, a robótica permite abarcar uma série de conteúdos a ensinar. Permite também aos intervenientes ter a noção dos desafios de construir um sistema completo.

A Universidade de Aveiro (UA) tem desenvolvido uma série de atividades de robótica com fins didáticos, das quais se destacam os concursos Micro-Rato e Ciber-Rato. Para o concurso Micro-Rato têm sido desenvolvidos vários robôs, quer pelas equipas participantes, quer por alunos nas suas dissertações de mestrado, quer mesmo pela própria organização da competição.

O concurso Micro-Rato, organizado desde 1995, é uma competição entre robôs autónomos e móveis que se desenrola num labirinto. Os robôs participantes devem ter dimensões menores que $300 \times 300 \times 400 \text{ mm}^3$. Num primeiro objetivo estes deslocam-se da área de partida até à área de chegada e num segundo objetivo devem regressar à área de partida ou perto dela, usando a informação que recolheram durante a execução do primeiro objetivo. A ação do concurso desenrola-se num labirinto em que os robôs devem evitar colidir com os obstáculos e com outros robôs [3].

Muitas vezes os participantes usam os robôs disponibilizados pela organização e têm um curto espaço de tempo para poderem trabalhar com eles. Por outro lado, uma vez que não são os seus robôs, as alterações que a estes se podem efetuar são limitadas e não há muita abertura à expansibilidade.

1.2 Objetivos

O objetivo desta dissertação de mestrado é a conceção, o projeto e a construção de uma plataforma robótica educacional que seja robusta e que tenha um conjunto base de funcionalidades que permitam a participação no concurso Micro-Rato. Podemos formalizar a seguinte lista de requisitos que pretendemos que a plataforma cumpra:

- Dimensões reduzidas;
- Baixo consumo de energia;
- Facilidade de construção;
- Suporte para futura expansão.

As dimensões reduzidas da plataforma permitirão o seu melhor manuseamento e transporte ao mesmo tempo que baixam o seu custo e consumo de energia. Uma vez que a plataforma é autónoma deve ter uma fonte de energia que se desloque com ela, daí que o baixo consumo seja

um objetivo para que esta tenha uma maior autonomia. A montagem da plataforma deve ser fácil de forma a que, conhecendo alguém o seu projeto, consiga, rapidamente levar a cabo a sua construção. Depois de construída a plataforma deve possibilitar uma fácil expansão e adição de funcionalidades, para que se abra um grande leque de aplicações.

Acresce, ainda, o desejo de que se construa um ambiente de teleoperação numa entidade externa ao robô, que possuindo melhores características de processamento seja capaz de alargar horizontes para as futuras aplicações deste. Esta entidade externa deve comunicar de uma forma sem fios com a plataforma.

1.3 Trabalho realizado

No âmbito desta dissertação de mestrado realizou-se todo o trabalho de conceção, projeto e construção de uma plataforma robótica educacional tentando que esta cumprisse com o que foi dito anteriormente. Foi também construído *software* de suporte que permite a futuros utilizadores terem uma camada de abstração que facilmente lhes permite interagir com a plataforma.

1.4 Organização da dissertação

Este documento está estruturado em 5 capítulos, o primeiro dos quais é este, a introdução. No capítulo seguinte abordar-se-á de forma mais exaustiva a questão da robótica educacional apresentando várias iniciativas que se podem encontrar um pouco por todo o mundo e algumas plataformas que serviram de inspiração para a que foi construída. Também se falará um pouco sobre as iniciativas de robótica educacional na UA. No Capítulo 3 irá apresentar-se o processo de conceção, projeto e construção da plataforma robótica analisando cada subsistema desta em particular e referindo como tudo se encaixa. De seguida, no Capítulo 4 apresentam-se os resultados que se obteve com esta dissertação de Mestrado. Por fim, conclui-se e apresenta-se o trabalho futuro no Capítulo 5.

Capítulo 2

Robótica na educação

A robótica usada como meio de transmissão de conhecimentos, permite ensinar não só conteúdos relacionados com a robótica em si, mas também com outras áreas do saber. O conjunto de atividades desenvolvidas por meio de robôs com o intuito de ensinar designa-se por robótica didática ou robótica educacional. Quando adicionalmente os robôs são móveis pode-se especificar dizendo robótica móvel educacional. No seguimento deste texto designar-se-á o conjunto destas atividades, seja por meio de robôs móveis ou não, por robótica educacional. Sendo uma atividade, por natureza, transversal, envolvendo diferentes campos do conhecimento, é possível usar a robótica para experiências educativas variadas.

A transversalidade da robótica pode ser atestada em [17], neste caso fala-se em específico do *RoboCup* [18] que é uma competição de robótica a nível mundial de que se falará mais à frente. As pesquisas das equipas envolvidas passam por áreas como a biologia, a sociologia, entre outras. São então possíveis experiências de aprendizagem amplas, sendo exemplo disso a disciplina *Autonomous Robotics* lecionada na *Case Western Reserve University* (CWRU) para a qual se inscrevem alunos de cursos das áreas da biologia, informática e ciências [19].

É possível distinguir entre dois usos da robótica educacional, seguindo, para isso, [20] e mantendo a designação em inglês atribuída aos dois grupos podem apresentar-se as seguintes designações:

- *Robots in Education* - Neste caso parte-se do princípio que a robótica é uma forma divertida de ensinar e usam-se robôs para transmitir conhecimentos úteis ao mesmo tempo que se divertem os utilizadores/alunos.
- *Robots for Education* - Uma vez que a robótica é uma área multidisciplinar e útil no processo de ensino, pode-se usar os robôs paralelamente ou substituindo outras ferramentas didáticas.

Na verdade e a propósito do primeiro caso, um dos grandes motivos que leva ao uso da robótica na educação, pelo menos a um nível pré-universitário é a sua componente lúdica, aumentando o interesse dos envolvidos nas tarefas. Este aumento de interesse e a diversão despoletada é comprovada pelos intervenientes nas atividades, e encontrado na bibliografia consultada [20], [19] e [17].

Contudo, o simples facto de ser divertido não é suficiente para motivar o uso da robótica em contextos educativos. É necessário comprovar a efetividade do ensino usando estes meios, sendo que por isso é mais comum encontrar motivações na forma do segundo item, pois é

mais fácil encontrar experiências comprovadas que levam a concluir que o uso da robótica é adequado [20]. Mesmo assim não se deve menosprezar o uso da robótica com o intuito de divertir os intervenientes, aliás ela é usada desta forma em muitos contextos.

Em termos da inclusão da robótica nos diferentes níveis de ensino pode fazer-se a distinção entre o ensino pré-universitário, e o ensino universitário e investigação. Num nível pré-universitário, a grande motivação para o uso da robótica é despertar o interesse dos envolvidos para conteúdos dentro das disciplinas que fazem parte do seu currículo e também captar futuros alunos para cursos universitários [17] nas áreas em que a robótica é usada como trabalho e/ou investigação. Já no ensino universitário e na investigação, o uso da robótica é mais no sentido de ser usada como uma ferramenta didática a par com outras, mas com a preocupação de saber claramente que conteúdos e com que objetivos é que se faz.

2.1 Ensino pré-universitário

Neste nível, a educação tende a ter um caráter geral, pelo que a robótica, salvo algumas exceções, tende a entrar como uma atividade extra-curricular. O seu uso espalha-se pelos vários níveis de ensino, desde o pré-escolar ao secundário com metas e objetivos diferentes.

Falando de cursos profissionais é possível encontrarmos exemplos do uso da robótica no ensino. Por exemplo em [21] é possível encontrar um artigo publicado no *website* da Escola Profissional de Braga sobre a sua participação no Festival Nacional de Robótica (FNR), na sua edição de 2012 onde conquistou o segundo lugar numa das provas [22]. Os alunos participantes eram provenientes de um curso de Técnico de Eletrónica, Automação e Comando. No currículo deste curso, que pode ser encontrado no *website* da escola, é possível ver que são contempladas disciplinas como Eletricidade e Eletrónica, Sistemas Digitais e Automação e Comando, a par com outras de formação específica e geral. Neste caso é possível usar um robô como veículo de ensino dos princípios base daquelas disciplinas, contudo levantam-se algumas questões ao fazê-lo. Veja-se um exemplo dado em [20]: para determinada aplicação os alunos tinham de usar a lei de *Ohm*, são então apresentadas quatro questões:

1. Será que os alunos no fim da tarefa sabem realmente a lei de *Ohm*?
2. Será que eles realmente percebem a lei (com todas as suas ramificações)?
3. O que é que aprendem mais da teoria de circuitos sem ser a lei de *Ohm*?
4. O que é que eles aprendem que não iam aprender se não estivessem com as mãos num robô?

Para responder às primeiras três questões pode-se submeter os alunos a exames regulares, trazendo isso mais trabalho para os professores que já tiveram de preparar as atividades robóticas. Já a quarta questão é a mais delicada de todas, uma vez que se os alunos aprendem mais com as mãos num robô deve-se intensificar o seu uso, contudo, se tal não é verdade a intensificação do seu uso vai levar a um maior trabalho por parte dos professores que não se vai refletir numa maior aprendizagem por parte dos alunos. A resposta a esta questão iria exigir uma recolha de dados ao longo do tempo, sobre versões com e sem robô da disciplina para que houvesse conclusões satisfatórias sobre os benefícios, ou a ausência deles, do uso da robótica na educação pré-universitária.

Já referida anteriormente a questão do tempo é muito importante nas atividades de robótica educacional, tanto para o lado dos alunos como para o lado dos professores. Segundo [20], conduzir uma disciplina recorrendo a robôs como ferramenta educacional, requer para além de mais tempo, mais espaço e mais pessoas. Como o tempo é um recurso muito limitado torna-se complicado para um professor conduzir experiências didáticas com robôs, ainda para mais quando não é de todo claro quais as vantagens de o fazer.

Apesar de toda a discussão anterior e apesar de se saber que um dos grandes motivos que leva ao uso da robótica a este nível é o divertimento que causa nos intervenientes, há várias vantagens que advêm do uso da robótica, não só a este nível como em níveis universitários que se irá abordar no início da secção seguinte.

2.2 Ensino universitário e investigação

No ensino universitário, a robótica encontra as mais variadas aplicações e finalidades. As vantagens de conduzir uma disciplina usando robôs passam, por exemplo pelo projeto que é considerado como a maior atividade de engenharia e não há dúvidas de que a robótica pode ser útil para a sua aprendizagem. Já o trabalho em equipa é considerado como a maior capacidade de um engenheiro e conduzindo atividades robóticas em grupo não há dúvidas que essa capacidade sai melhorada [20].

De certa forma, em disciplinas introdutórias, é óbvio haver uso da robótica uma vez que permite aos alunos aprenderem, de uma forma lúdica, conceitos básicos, seja de teoria de circuitos, introdução ao controlo ou de programação [20]. Ao mesmo tempo as suas capacidades de projeto e trabalho em grupo saem beneficiadas.

Contudo, a robótica educacional não se fica só pelas introduções a cursos superiores. Em [20] é referida também a possibilidade de usar robôs nas áreas da programação, em disciplinas relacionadas com a inteligência artificial. Neste campo a robótica proporciona um meio de teste para os algoritmos desenvolvidos, com a vantagem de ter a imprevisibilidade e a complexidade de um meio físico real e não simulado, sendo que por vezes algumas das características de um meio real não estão previstas, ou são esquecidas, nos modelos teóricos com que se lida.

Nos cursos de engenharia há alguma dificuldade na transição do mundo universitário para o mundo industrial. Por vezes tende-se a negligenciar o facto de que, no segundo mundo, as soluções que se desenvolvem para sistemas, sejam de *hardware* ou de *software*, vão ser apenas uma parte de um sistema maior em que o projeto deve ser feito de forma a visar a integração entre os vários módulos, desenvolvidos por diferentes equipas. A robótica pode ser usada como forma de manter os alunos, futuros engenheiros, alerta e preparados para enfrentar os desafios de desenhar apenas uma parte de um sistema, tendo depois de a integrar no todo. A experiência na disciplina *Autonomous Robots* lecionada na CWRU [19], já referida anteriormente, pretendia familiarizar os seus alunos com estas situações. De início era-lhes pedido que desenhassem componentes individuais de um sistema robótico. Posteriormente, era pedida a integração dos vários módulos que, de propósito, não iria funcionar: os alunos eram divididos em grupos e cada grupo deveria construir um robô, como cada um dos elementos era deixado a projetar uma parte sem saber que tipo de requisitos deveria obedecer para a sua integração esta, normalmente, falhava. Depois desta primeira experiência eram incentivados a trabalhar como um grupo tendo em conta os requisitos a que o seu módulo deveria obedecer. Para o futuro cada um deles ganhava a perceção do que significa desenvolver um sistema e

iam estar atentos aos requisitos de integração.

Há uma experiência citada em toda a bibliografia e que é apontada como precursora do uso da robótica na educação. Trata-se da disciplina 6.270 do *Massachusetts Institute of Technology* (MIT) [23], iniciada nos finais dos anos 80 é considerado como o precursor do uso da robótica na educação. É uma disciplina que decorre ao longo de um semestre e os seus alunos constroem um robô para participar numa competição no final da disciplina. Todos os anos existe um desafio novo baseado numa história em que os robôs vão ser protagonistas. Estes devem ser capazes de navegar, reconhecer oponentes e manobrar objetos num campo desconhecido à priori. Pretende-se que os alunos adquiram conhecimentos de base de projeto de robôs.

A investigação em robótica é uma atividade que já se situa um pouco fora do âmbito desta dissertação, uma vez que as experiências de robótica educacional referidas são orientadas para serem um meio de ensino em vez de serem o objeto de estudo, como acontece com a investigação em robótica. Contudo, pode-se aplicar o uso de robôs para fazer investigação noutras áreas, como a inteligência artificial, já referida. Nos campos das neurociências o uso de robôs também pode ser útil para lembrar os investigadores da complexidade do funcionamento do cérebro humano, fazendo-os ver a dificuldade de implementar algoritmos de decisão e controlo em robôs [19]. Na área da sociologia os robôs podem ser usados para entender comportamentos sociais ou individuais ou para testar modelos de comportamentos [17]. Por fim, como último exemplo, na biologia os robôs podem ser usados para testar teorias de evolução com algoritmos genéticos. Estes algoritmos consistem numa série de comportamentos que são agrupados numa série de programas, estes são avaliados e alguns deles mantidos para testar numa nova geração de algoritmos gerados [17].

2.3 Concursos de robótica

Em concursos de robótica, seguindo determinadas regras e com um objetivo comum avalia-se o desempenho de várias equipas ou indivíduos. É importante realçar que estes concursos podem ser também usados com propósitos didáticos adequando os seus objetivos e regras para que os participantes sejam impelidos a aprender conteúdos de determinadas áreas. Para além disto, são uma atividade em que é proporcionada diversão e competição.

Os concursos de robótica realizados no meio académico mantêm os alunos motivados e interessados no trabalho que vão desenvolvendo, uma vez que se cria espírito competitivo e vontade de vencer. Existe até a possibilidade de fazer de um concurso um método de avaliação dos alunos das disciplinas em que se faz uso educacional de robôs. Neste caso, há questões de sabermos se este método é justo para avaliar, se pode substituir outros métodos de avaliação, se os pode complementar e de que forma. Num concurso de robótica é comum duvidar-se se os alunos/concorrentes realmente entenderam aquilo que fizeram ou se por um método de tentativa-erro chegaram a uma solução para o problema, solução essa que não entendem. Assim, o facto de terem uma boa classificação no concurso, poderia conduzir a uma avaliação injusta. É necessário haver um correto balanceamento das várias componentes de avaliação [20]. No exemplo da CWRU já falado e na disciplina 6.270 do MIT, o concurso é tido em conta na avaliação dos alunos. É também importante dizer que embora a motivação extra que a competição possa trazer seja importante, também há quem defenda que usar uma exposição como avaliação dos alunos é mais vantajoso [20].

Não sobrevalorizando estas atividades, elas são muito interessantes pela motivação que

induzem nos participantes e pelo facto de poderem juntar um grupo de várias pessoas em torno de um objetivo comum, contribuindo para o enriquecimento da sua capacidade de trabalhar em grupo.

Nas sub-seções seguintes apresentam-se algumas destas iniciativas, escolhidas de entre as demais por representarem algum interesse para este documento e para o trabalho que foi efetuado nesta dissertação e por ter sido mais fácil reunir informação sobre elas. De qualquer forma, pode-se consultar o *website* referido em [24] onde se pode encontrar uma extensa lista de competições de robótica realizadas um pouco por todo o mundo.

2.3.1 Trinity College Fire Fighting

Este concurso é realizado pelo *Trinity College* de Hartford nos Estados Unidos. Tem vários públicos alvo dependendo das atividades em questão, e é aberto a todo o mundo, com competições em equipa e individuais [25].

É composto por vários tipos de atividades em volta da extinção de focos de fogo (simulados por uma vela e reais) e realização de tarefas domésticas.

No que ao fogo diz respeito, quando os focos são simulados por velas, a extinção faz-se numa área que simula uma casa e o robô deve localizar os focos de forma autónoma e apagá-los. É possível ver na Figura 2.1, um exemplo do ambiente em que este concurso se dá. Em níveis de dificuldade mais baixos, júnior e *high-school*, é dada a planta da casa, já para níveis mais difíceis a localização das portas entre as divisões pode variar e desconhece-se o número de focos sendo que todos têm de ser extintos.

Para tarefas domésticas existe a modalidade RoboWaiter em que o objetivo é transportar autonomamente um prato de comida para uma mesa de uma maneira confiável e eficiente, o contexto em que ocorre a prova pretende simular uma casa com dois bonecos simulando pessoas desabilitadas.

Muito importante é o facto de os robôs deverem ser construídos pelos participantes o que coloca a fasquia de participação muito alta e aumenta o desafio do concurso.

Na Guarda, em Portugal, existe uma competição baseada nesta, designada por Robô Bombeiro [26].



Figura 2.1: Interior de uma casa, simulado para a realização de uma prova do concurso robô bombeiro

2.3.2 RoboCup

O RoboCup é uma competição de robótica a nível mundial que tem um objetivo que em muito se deve à derrota de Gary Kasparov, campeão mundial de xadrez, frente ao computador *IBM Deep Blue* [27] em 1997. Este feito levou à definição de um projeto com um objetivo ambicioso que é formalizado da seguinte forma de acordo com o *website* desta iniciativa [18]:

“Em meados do século XXI, uma equipa de robôs futebolistas humanóides e completamente autónomos deve ganhar o jogo de futebol, de acordo com as regras oficiais da FIFA, contra o vencedor do mais recente campeonato do mundo”

Esta é uma iniciativa massiva que reúne milhares de participantes todos os anos em torno da disseminação da robótica, da investigação em robótica e mais recentemente em torno da educação. Há uma iniciativa especial vocacionada para pessoas até aos 19 anos designada RoboCup Junior [28]. Segundo o *website* da iniciativa ela é definida como sendo orientada ao projeto, focada em providenciar aos participantes uma experiência *hands-on* onde podem aprender, ao mesmo tempo que a sua curiosidade é estimulada e aumenta o seu conforto com a tecnologia. Foi primeiro apresentada em 1998 em Paris e no ano seguinte realizou-se um *workshop* em Estocolmo. Foi realizada pela primeira vez em Melbourne reunindo 100 crianças. Na edição de 2011 participaram 289 equipas de 29 países o que mostra um grande crescimento e o interesse que esta iniciativa tem.

Do ponto de vista didático há vários desafios que se colocam que de certa forma já foram abordados anteriormente, prendendo-se principalmente com a dificuldade dos professores integrarem a robótica nos currículos escolares e com a formação adicional que lhes é necessária para usarem robôs. Por outro lado o trabalho que os alunos têm de fazer envolve *software* e *hardware* que são de difícil desenvolvimento para o seu nível de escolaridade e torna-se complicado fazer com que o façam de forma reutilizável sem a ajuda dos professores, o que obriga a que ao longo dos anos o desenvolvimento seja lento, uma vez que cada equipa tem de construir os seus próprios recursos de raiz. Mesmo assim tem-se verificado, como já foi dito, um crescimento no número de participantes.

Esta iniciativa compreende três modalidades principais que se irá falar de seguida com mais pormenor, sendo elas: futebol júnior, busca e salvamento júnior e dança júnior. Em Portugal estas modalidades fazem parte do FNR [22] que é a maior competição de robótica a nível nacional, envolvendo não só os participantes das três modalidades referidas como também participantes oriundos do ensino superior. Serve também como prova eliminatória para as iniciativas RoboCup anuais onde se realizam as competições do RoboCup Júnior.

Futebol júnior

A modalidade futebol júnior consiste em equipas de 2 robôs que jogam numa área limitada e fechada e tentam marcar golos introduzindo uma bola emissora de infravermelhos na baliza do adversário.

O campo de jogo tem uma dimensão de $122 \times 183 \text{ cm}^2$ e está coberto com um gradiente de escala de cinza que permite aos robôs orientarem-se. As balizas têm tamanhos variáveis para duas sub-modalidades diferentes: *Soccer A*, $45 \times 14 \text{ cm}^2$ e *Soccer B*, $60 \times 10 \text{ cm}^2$. Em ambos os casos, as balizas estão pintadas de cor diferente em cada lado, amarelo e azul.

Não há limites ao tamanho das equipas, a única condição imposta é que sejam compostas

por mais do que um participante, sendo que os participantes e os robôs não podem ser partilhados por equipas diferentes.

Não há muitas restrições à construção dos robôs, estas colocam-se apenas nas suas dimensões e tensões de alimentação. O diâmetro e altura não devem exceder 22 cm e o peso e a tensão de alimentação não devem exceder, respetivamente, 1100 g e 12 V para a liga mais leve, *Light Weight League*, e 2400 g e 15 V para a liga mais pesada *Open League*.

Pode haver comunicação entre os robôs desde que seja efetuada usando tecnologia *Bluetooth*, classe 2 ou 3, com alcance menor que 20 m ou *ZigBee*, contudo não é garantido que os canais necessários para a comunicação estejam disponíveis. Qualquer outra forma de comunicação é proibida.

Na Figura 2.2a é possível ver dois robôs participantes próximos de uma das balizas.

Busca e salvamento júnior

Esta modalidade tem como objetivo identificar vítimas em cenários de desastre simulados, que podem ir desde seguir uma linha numa superfície plana a negociar caminhos contornando obstáculos em terreno irregular.

Há duas submodalidades diferentes *Rescue A* e *Rescue B*. No caso da *Rescue A* as vítimas são assinaladas por latas de alumínio que devem ser movidas para uma *safe zone*, enquanto que no caso da *Rescue B* as vítimas são assinaladas no chão por meio de um material emissor de radiação infravermelha.

Tal como no caso anterior as equipas devem ter mais do que uma pessoa e os participantes e os robôs não podem ser partilhados por equipas diferentes.

Os robôs são de construção livre desde que a sua altura não exceda 300 mm. A comunicação é permitida nos mesmos moldes da modalidade anterior.

Na Figura 2.2b é possível ver um exemplo desta modalidade em que duas garrafas cheias de areia obstruem a passagem através do seguimento da linha preta.

Dança júnior

A dança júnior é uma modalidade em que robôs e humanos dançam, harmoniosamente, ao som de uma música usando roupas criativas. O desafio desta modalidade é fazer com que os robôs dançam de forma síncrona com a música. Porém a forma como isso é feito não tem de ser necessariamente reativa, ou seja, o robô reagir à música.

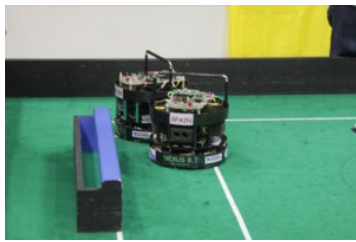
Há duas sub-modalidades: teatro e dança. Na primeira os participantes usam os robôs para contar uma história suportada pela música e na segunda apenas executam uma coreografia pré-programada.

Os critérios de avaliação das equipas passam não só pela parte técnica, como também pela beleza da performance criada ou mesmo pela escolha das roupas.

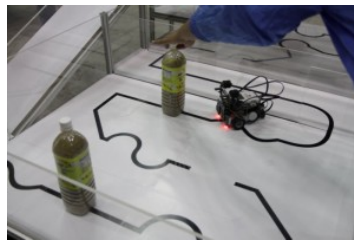
Mais uma vez as equipas devem ter mais de um elemento e os participantes e os robôs não podem ser partilhados por equipas diferentes.

A construção de robôs é livre, contudo uma altura superior a 4 m contando do palco deve ser discutida com o júri.

Na Figura 2.2c é possível ver um momento de uma das coreografias, nesta competição assiste-se normalmente a espetáculos belos do ponto de vista artístico embora o desafio técnico seja menor que o dos anteriores.



(a) Futebol Júnior, fonte [28]



(b) Busca e salvamento júnior, fonte [28]



(c) Dança júnior, fonte [22]

Figura 2.2: As várias modalidades do RoboCup Júnior

2.3.3 Mobot Races

Este concurso [1] realiza-se na *Carnegie Mellon University* (CMU) em Pittsburgh, nos Estados Unidos. Tem a particularidade de se realizar num ambiente exterior ao “ar livre” com um piso asfaltado e desnivelado, este tipo de ambiente coloca um desafio acrescido para o funcionamento dos robôs pela sua imprevisibilidade e pela robustez física extra que exige na sua construção. A somar ao que já foi dito, o caminho que o robô tem de percorrer tem zonas de relva a ladear e obstáculos fixos.

Na sua essência o objetivo deste concurso é que um robô siga uma linha branca, desenhada no asfalto ao longo de um caminho pré-determinado, podendo haver cruzamentos com outras linhas.

Os robôs podem ser desenhados pelos participantes sendo que a única restrição é que sejam capazes de passar através dos pórticos de 46 cm de lado como o que se pode ver na Figura 2.3. Contudo a organização do evento disponibiliza *kits* robóticos chamados *Mobot Kit* que os alunos podem usar para assemblar o seu próprio robô, fazem parte deste *kit* um *chassis* mecânico e alguns sensores, bem como componentes eletrónicos de apoio e um livro de introdução à eletrónica.



Figura 2.3: Pórtico na pista das Mobot Races, fonte [1]

2.3.4 MicroMouse

O MicroMouse é um concurso disseminado um pouco por todo o mundo. As regras são de certa forma similares, trata-se de um pequeno robô, “MicroMouse”, que busca num labirinto pelo ponto de chegada.

A origem remonta a 1977 quando a revista *IEEE Spectrum* lançou as especificações do concurso como “*The Amazing MicroMouse Contest*” que seria realizado em 1979 em Nova Iorque, envolvendo 15 concorrentes. Nos anos seguintes a competição passou para a Europa: em 1980 em Londres com 9 participantes e em 1981 em Paris com 13 participantes. Na sua primeira versão era requerido a resolução de um labirinto de 3×3 m². Os labirintos que os robôs têm de resolver nesta competição são muito parecidos com o que podemos ver na Figura 2.4. Pode-se encontrar esta e mais informação em [29] e [30].

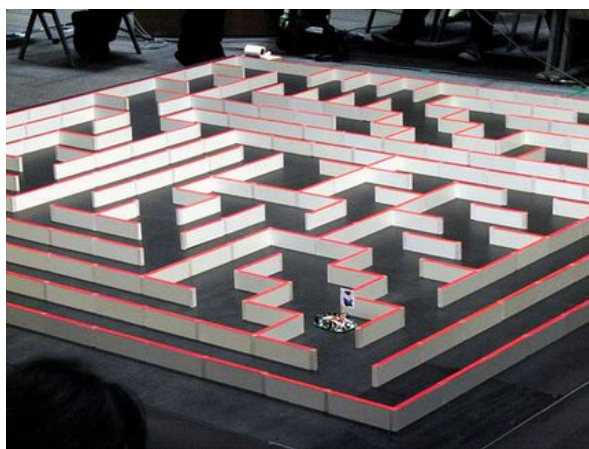


Figura 2.4: Labirinto do MicroMouse, fonte [2]

Em 2011 foi batido o recorde para o robô MicroMouse mais rápido que resolveu o labirinto do *All Japan MicroMouse Contest* em 3,921 s num labirinto similar ao da Figura 2.4 [31].

As realizações desta competição podem ser um concurso de robótica *per si* ou podem fazer parte de um evento maior. No caso do Japão realiza-se anualmente uma atividade chamada *All Japan MicroMouse Contest* [32] que é totalmente dedicada ao concurso MicroMouse. Já no Reino Unido realiza-se em Birmingham outra competição completamente dedicada a este concurso [30]. Na Polónia, realizam-se em Varsóvia [2] e em Rybnik [33] concursos de robótica que incluem competições MicroMouse. Em Portugal existe na Universidade de Trás-os-montes e Alto Douro (UTAD) uma iniciativa chamada *Micromouse Portuguese Contest* também dedicada ao concurso MicroMouse [34].

Na UA existe uma iniciativa chamada Micro-Rato que toma inspiração do concurso MicroMouse, contudo, com algumas diferenças. Na secção seguinte ir-se-á abordar a robótica educacional na UA e falar mais em concreto deste concurso.

2.4 Robótica educacional na Universidade de Aveiro

Na UA têm-se desenvolvido várias atividades de robótica educacional, não só orientadas para os seus alunos, como também, orientadas a alunos de outras universidades ou de outros níveis de ensino.

O concurso Micro-Rato é a iniciativa mais antiga, quer na UA, quer no país. Foi criado em 1995 e tem incentivado uma série de atividades de robótica educacional organizadas pelo Departamento de Eletrónica Telecomunicações e Informática (DETI). Já abordado também na introdução deste documento, este concurso tem também servido de ponto de partida para várias dissertações ou artigos científicos. Vamos então começar por explicar em mais pormenor o que é o Micro-Rato e o Ciber-Rato, que é um concurso que foi inspirado no primeiro.

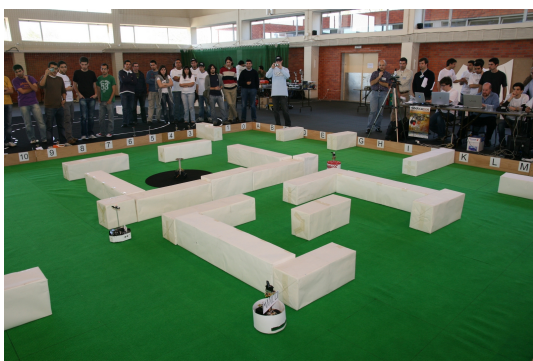
2.4.1 Concurso Micro-Rato e Ciber-Rato

Criado em 1995 inspirando-se no concurso MicroMouse este concurso tem-se realizado quase ininterruptamente com um considerável número de participantes todos os anos.

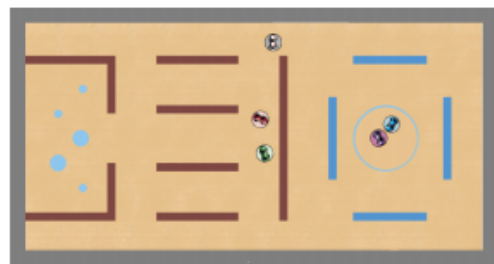
Os robôs intervenientes têm, como já foi dito, de navegar num labirinto que desconhecem à priori, orientando-se para um farol de luz infravermelha. No caminho devem evitar obstáculos e outros robôs que também podem estar a concorrer. A zona de finalização estará assinalada por meio de uma superfície não refletora de luz infravermelha, ao passo que todo o resto do labirinto terá uma superfície que reflete este tipo de luz. No fim de cumprirem este objetivo podem ainda tentar o segundo que é o regresso ao ponto de partida, com recurso à informação armazenada durante a ida para o farol.

O concurso Ciber-Rato surgiu como uma versão simulada do concurso Micro-Rato, primeiro com o objetivo de servir de teste para os algoritmos de resolução dos labirintos. Neste momento este concurso funciona como uma competição *per si* em que agentes robóticos competem num ambiente simulado. Os objetivos são semelhantes aos do concurso Micro-Rato.

Nas Figuras 2.5a e 2.5b é possível ver exemplos dos labirintos em que estes dois concursos se dão, para o concurso Micro-Rato e Ciber-Rato respetivamente. Na imagem da esquerda referente ao concurso Micro-Rato o ponto de partida localiza-se junto aos limites de madeira da arena e o ponto de chegada é assinalado pelo círculo preto que tem no seu centro o farol emissor de luz infravermelha. Na imagem da direita é possível notar também o ponto de partida e de chegada no concurso Ciber-Rato. Em qualquer um dos casos existem obstáculos ao avanço dos agentes robóticos, sendo que alguns deles podem obstruir a visão para o farol o que dificulta a orientação do robô.



(a)



(b)

Figura 2.5: Exemplos de labirintos dos concursos Micro-Rato [35], à direita, e Ciber-Rato [36] à esquerda

No caso do Micro-Rato os participantes podem construir o seu próprio robô, ou requerer um à organização. Estes robôs que a organização disponibiliza foram sendo construídos

ao longo do tempo e alguns deles serão abordados na secção 2.5. No caso de construir o seu próprio robô, os participantes devem saber que as dimensões deste estão limitadas a $300 \times 300 \times 400 \text{ mm}^3$ da forma que é indicada na Figura 2.6. A locomoção deve ser apenas baseada em motores elétricos e a alimentação feita por meio de baterias e qualquer comunicação entre robôs ou entre robôs e entidade externa é proibida. O robô deve possuir dois botões acessíveis, *START* e *STOP*, de modo a poder atuar rapidamente no seu comportamento e deve possuir um indicador luminoso para assinalar que chegou ao fim do primeiro objetivo ou ao fim da prova.

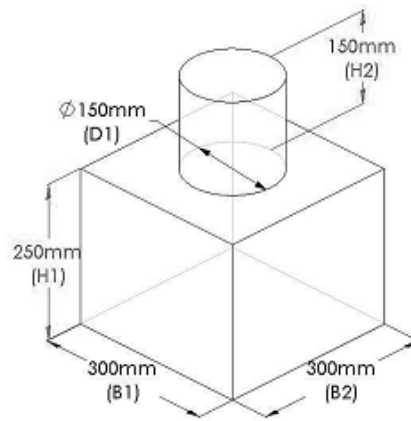


Figura 2.6: Limites máximos dos robôs participantes, fonte [3]

A prova tem uma duração máxima de 4 min, destes, o cumprimento do primeiro objetivo tem um limite máximo de 3 min. Entre vários critérios que determinam a classificação de uma equipa, estão: critérios temporais, penalizações acumuladas e, tendo tentado o segundo objetivo, a distância entre o ponto em que o robô se imobilizou e o ponto de partida [3].

As equipas podem ter qualquer número de elementos.

Para o concurso Ciber-Rato as regras são uma versão adaptada destas para um ambiente simulado. Nesta modalidade os agentes robóticos são dotados de mais sensores do que aqueles que as plataformas cedidas pelo DETI normalmente têm.

Estes dois concursos não têm sido as únicas iniciativas que o DETI tem levado a cabo.

2.4.2 Outras iniciativas de robótica educacional

Uma outra iniciativa levada a cabo foi a exposição Sítio dos Robôs e a Oficina dos Robôs presentes na Fábrica Centro Ciência Viva de Aveiro [37].

Na exposição Sítio dos Robôs é feita uma introdução progressiva do público no mundo da robótica autónoma e móvel. Numa primeira parte, são-lhes mostradas algumas demonstrações de bom efeito visual para cativar o seu interesse. Estas demonstrações podem ser, por exemplo: robôs que se movem numa sequência pré-programada ou robôs que apenas tomem alguma ação mediante interação com o público. Depois de motivado, ao visitante, são então explicados os componentes básicos da construção de robôs móveis e autónomos, os sensores, os atuadores e comportamentos básicos. Após entendidos os conceitos básicos da robótica podem construir o seu próprio robô na Oficina dos Robôs.

Contudo, a participação do DETI em atividades de robótica educacional vai mais além. São também levadas a cabo iniciativas no âmbito da Academia de Verão que se realiza todos os anos no final do ano letivo, bem como atividades enquadradas na semana aberta da ciência e tecnologia. Tem-se também participado na divulgação do departamento pelas escolas secundárias, por vezes por meio de algumas atividades de robótica educacional.

2.5 Plataformas robóticas

Nas duas secções anteriores abordámos várias iniciativas de robótica educacional quer espalhadas pelo mundo quer perto de nós. Nesta secção vai-se abordar um dos principais meios para estas iniciativas, as plataformas robóticas, tendo em conta a sua importância no âmbito desta dissertação.

Vamos começar por abordar as plataformas criadas no DETI que têm sido desenvolvidas ao longo dos anos, muito graças ao concurso Micro-Rato e com o objetivo de permitir aos alunos do DETI a participação no mesmo, como foi referido na introdução. Contudo, o desenvolvimento de plataformas robóticas foi para além do concurso e permitiu a sua inclusão nas atividades que foram referidas na secção anterior. Vai-se também abordar plataformas existentes no mercado e que são semelhantes àquilo que pretendemos.

O tipo de robótica pertinente para esta dissertação é a robótica móvel e autónoma. Como já se percebeu a plataforma que se pretende construir está muito relacionada com iniciativas de robótica educacional, mas principalmente, com iniciativas que, enquadrando-se neste tipo de robótica, possuem ao mesmo tempo um carácter móvel e autónomo. Neste tipo de robótica os robôs movem-se e tomam decisões sem qualquer intervenção humana.

Para se ter um agente robótico autónomo, este necessita essencialmente de três blocos constituintes principais. Necessita de recolher dados do ambiente que o rodeia, necessita de processar esses dados e necessita de agir. Assim este tipo de robôs necessita de: sentir, pensar e agir. Como se está a falar de robótica móvel, a componente de agir implica também que se possa agir sobre a posição do robô no espaço. Adicionalmente, e cada vez mais importante, a comunicação faz também parte deste paradigma. Entre os dados que se podem comunicar estão, por exemplo, dados sensoriais [38].

2.5.1 Primeira plataforma Micro-Rato

Esta plataforma foi criada em 2003 para permitir aos alunos, sem construir o seu próprio robô, a participação no concurso Micro-Rato. Assim podiam-se concentrar mais nos desafios do *software* de controlo.

Na Figura 2.7 pode-se observar várias vistas da plataforma e pode-se ver os seus constituintes básicos que conformam com as regras que já foram referidas anteriormente. Tratava-se de uma base de madeira que servia de suporte para os vários componentes da plataforma que se encaixavam nesta.

Interessa referir o que o robô possuía em termos do paradigma referido: sentir, pensar, agir. Assim vamos abordar de seguida estes itens.

Suporte sensorial

O suporte sensorial deste robô era constituído por vários tipos de sensores que permitia ter uma perceção do meio suficiente para cumprir os objetivos do concurso, ou pelo menos o

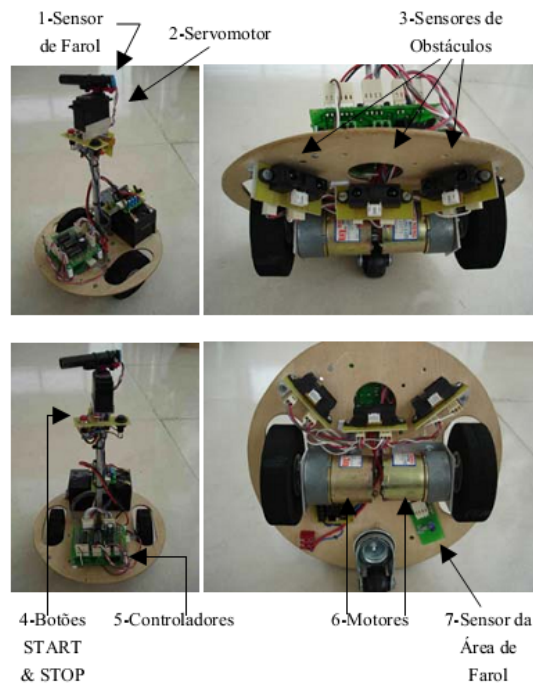


Figura 2.7: Plataforma Micro-Rato, fonte [4]

primeiro deles que era navegar até ao farol.

- A **Deteção de obstáculos** era feita por meio de três sensores Sharp GP2D12, estes sensores permitem detetar obstáculos refletores de luz infravermelha numa gama de distâncias de 10 a 80 cm à frente do robô. Na Figura 2.7 estes sensores estão assinalados pelo número 3.
- A **Deteção de farol** era feita por meio de um recetor de infravermelhos. Devido às características da onda emitida pelo farol que localiza o ponto de chegada, este recetor era o mesmo que é usado nas televisões para receber o sinal do comando. Este sensor assenta num servo-motor que faz um varrimento de 180° permitindo saber a direção em que o farol se encontra sendo que este conjunto está assinalado pelos números 1 e 2 na Figura 2.7.
- A **Deteção da zona de farol** permitia saber ao robô que havia cumprido o primeiro objetivo da prova, ou seja, tinha atingido o farol de infravermelhos. Este era assinalado por meio de uma zona não refletora no chão por oposição a todo o resto do labirinto em que o chão refletia luz infravermelha. O robô usava um sistema baseado num par emissor e recetor de luz infravermelha como pode ser visto no número 7 da Figura 2.7. Quando o chão não refletia, ou seja, estava-se na área do farol, o recetor deixava de receber luz infravermelha.
- Para conformar com as regras, estavam presentes no robô, como indicado no número 4 da Figura 2.7 os **Botões START e STOP** que permitiam marcar o início da prova do robô, ou a imobilização do mesmo.

Atuadores

Os atuadores presentes nesta plataforma permitiam a sua locomoção bem como outras funções necessárias ao robô que serão enumeradas de seguida.

- A **Locomoção** do robô era feita de um modo diferencial com dois motores elétricos DC colocados como é possível ver na Figura 2.7, no número 6. No caso deste robô o controlo dos motores era feito em malha aberta uma vez que não havia forma de saber que distância é que o robô havia andado. Desta forma, a conclusão do segundo objetivo da prova, o regresso à origem, era dificultado ao mesmo tempo que também era dificultado o controlo do próprio sistema de locomoção: uma vez que dois motores nunca são iguais, mesmo sendo de modelos iguais e a mesma tensão aplicada a motores diferentes produz velocidades diferentes.
- O **Movimento do sensor de farol** era conseguido por meio de um servo-motor que permitia, como já foi dito, fazer um varrimento de 180°. O conjunto servomotor e detetor de farol era colocado a uma altura que permitisse estar em linha de vista com o farol, o que levava ao aumento da altura global do robô.
- Para poupança de energia e para que a luz infravermelha dos sensores de obstáculos não prejudicasse os outros robôs intervenientes na prova era possível desligá-los.
- Para sinalizar as várias etapas da prova, o robô possuía um **Light Emitting Diode (LED) indicador**.

Suporte de processamento

Na Figura 2.7 no número 5, é possível ver a placa de controlo desta plataforma. Esta placa possuía em primeiro lugar todas as interfaces que permitiam fazer o *drive* dos periféricos (sensores e atuadores) e a unidade de controlo.

O controlo era implementado em software num microcontrolador PIC da Microchip, em específico o microcontrolador PIC16F876 [4] com arquitetura de 8 bits. Este microcontrolador possui 8 kWords de memória de programa, 368 bytes de memória RAM, vários periféricos, desde *timers* e *Analog to Digital Converter* (ADC) a interfaces e comunicações com o exterior. É capaz de executar até 5×10^6 instruções por segundo, trabalhando a 20 MHz.

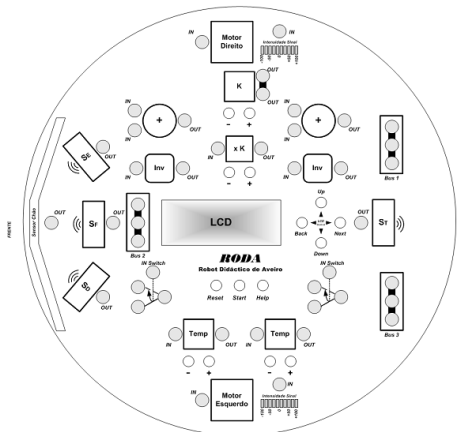
Recentemente esta placa de controlo foi substituída por outra em que os *drivers* dos periféricos foram atualizados e o microcontrolador passou a ser um PIC32.

2.5.2 RoDA

A plataforma RoDA partiu do conceito de ensinar princípios básicos de robótica móvel e autónoma emulando um método de programação *hard-wire* em que se podiam interligar vários módulos constituintes de um sistema de controlo de modo a implementar um certo comportamento. Na Figura 2.8a é possível ver o esquema da placa superior desta plataforma em que se podia implementar o controlo interligando os vários blocos. Era possível adaptar a plataforma para o concurso Micro-Rato, bastando para isso substituir a placa superior por uma placa especialmente adaptada para o concurso Micro-Rato. Como se pode ver na figura 2.8b, trata-se de uma plataforma com três bases que servem de suporte a toda a estrutura elétrica.

Em termos de sensores e atuadores, estes eram muito similares aos presentes na plataforma referida anteriormente com algumas diferenças de construção. Era também possível a deteção de obstáculos atrás do robô e havia mais sensores de infravermelhos apontados ao chão, sendo assim possível detetar com mais pormenor zonas refletoras e não refletoras. Esta plataforma possuía também, acoplados aos motores de locomoção, *encoders* que permitiam saber qual a distância percorrida pelo robô, desta forma era facilitado o cumprimento do segundo objetivo (o regresso ao ponto de partida) e eram corrigidos os problemas enumerados anteriormente quanto à locomoção, permitindo um controlo em malha fechada da velocidade dos motores. Quando em modo Micro-Rato, podia-se dispensar algum excedente de sensores montando apenas os necessários.

Já a unidade de processamento era baseada num microcontrolador PIC18F4580 de 8 bits que por comparação com o presente na plataforma anterior, antes do controlo ser convertido para PIC32, apresentava mais memória quer em termos de programa quer RAM, 16 kWords e 1536 bytes respetivamente, mais periféricos e uma maior frequência de trabalho: 10 MHz.



(a) Placa de ligações virtuais, fonte [39] (b) Plataforma RoDA completa, vista de frente, fonte [39]

Figura 2.8: Plataforma RoDA

Esta plataforma permitia uma série de experiências interligando os blocos presentes na placa superior. Era possível ir de comportamentos simples, como ser atraída por um obstáculo a comportamentos mais complexos, como andar em cima de uma mesa sem cair.

2.5.3 Pololu 3 π

A plataforma robótica Pololu 3 π [40] é uma pequena plataforma que toma o nome do seu diâmetro, próximo de 3 π , 9,4 cm, e está disponível no mercado por 73,75 EUR. Na Figura 2.9 é possível ver esta plataforma em duas vistas legendadas em inglês.

Nesta plataforma, uma *Placa de Circuito Impresso* (PCB) é ao mesmo tempo o suporte físico. É nesta placa que estão localizados todos os sensores e atuadores do robô, bem como a sua unidade de controlo.

Uma das características desta plataforma é a sua expansibilidade, sendo possível comprar um conjunto de acessórios que permitem adicionar novas funcionalidades à plataforma. Desta forma temos acesso a uma plataforma mais acessível em termos de preço e podemos

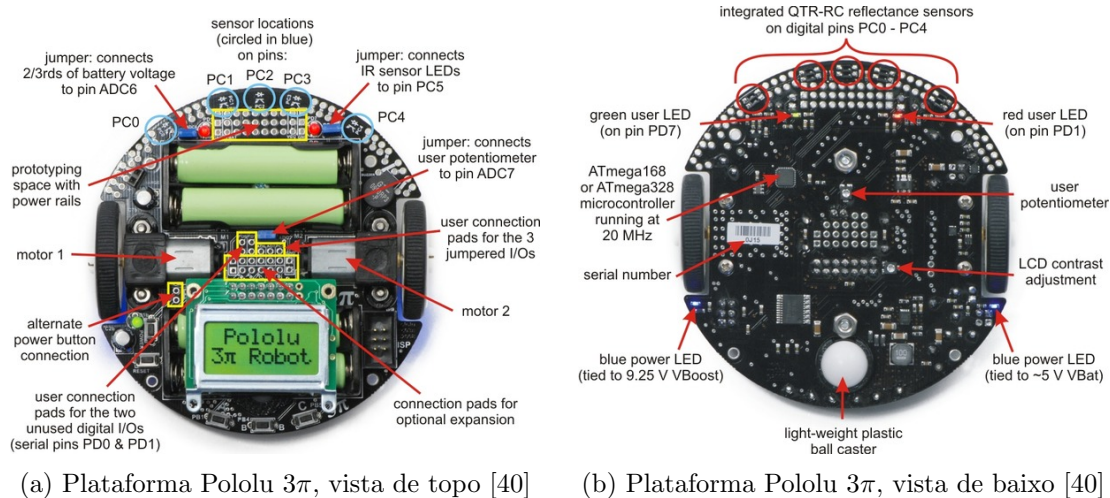


Figura 2.9: Plataforma Pololu 3 π

posteriormente customizá-la.

De seguida far-se-á uma análise do que esta plataforma possui em termos de suporte sensorial, atuadores e suporte de processamento.

Suporte sensorial

Este robô possui Sensores de refletância implementados através de vários sensores individuais compostos de um par emissor e recetor de luz infravermelha. Assim, é possível discriminar uma linha não refletora num fundo refletor ou vice-versa. Na Figura 2.9b é possível ver estes sensores assinalados como *Integrated QTR-RC reflectance sensors*. Faz também parte deste robô um conjunto de **Botões e um potenciômetro**, o que permite alguma interface com o utilizador.

Atuadores

À semelhança das plataformas anteriores também esta possui um sistema de locomoção diferencial assente em dois motores DC. Por ser de reduzidas dimensões os motores acabam por ser também reduzidos. Neste caso uma caixa redutora com uma conversão baixa 30:1 leva a uma velocidade de rotação máxima de 700 rpm o que associado ao diâmetro das rodas de 2,4 cm conduz a uma velocidade máxima de aproximadamente 88 cm/s o que é bastante rápido.

A plataforma possui também LEDs e um ecrã *Liquid-crystal display* (LCD) para interface com o utilizador.

Suporte de processamento

A unidade de processamento deste robô é composta, como é indicado na Figura 2.9b por um microcontrolador ATmega168 ou ATmega328, ambos produzidos pela ATMEL. Ambos são processadores de 8 bits com uma frequência máxima de operação de 20 MHz. O primeiro tem 16 kbyte e o segundo 32 kbyte de memória de programa. Já em termos de memória RAM têm 1 kbyte e 2 kbyte, respetivamente.

2.5.4 EVALBOT

A plataforma EVALBOT foi desenvolvida pela Texas Instruments. É uma plataforma com uma forma redonda com 10,16 cm de diâmetro e disponível no mercado a partir de 121,65 EUR. É também uma plataforma em que uma PCB é ao mesmo tempo o suporte físico do robô.

Esta plataforma possui um conjunto de dois conectores destinados a adicionar comunicação sem fios à plataforma. Para além disso o suporte para futuras expansões é reduzido [5].

Na figura 2.10 é possível observar esta plataforma. Pode-se observar as diferentes interfaces de comunicação do lado direito da figura, o microcontrolador no centro e um *speaker* à frente. A plataforma vem equipada com um *codec* de áudio e um *display*.



Figura 2.10: Plataforma robótica EVALBOT [5]

Suporte sensorial

O suporte sensorial desta plataforma é muito reduzido, possui dois sensores de toque, e *encoders* opto-eletrónicos que permitem medir a rotação angular com 45° de precisão. Para além disso, possui também botões para interação.

Atuadores

Em termos de atuação, a plataforma possui dois motores e um sistema de locomoção diferencial. Possui ainda alguns LEDs e o *display* já referido.

Suporte para processamento

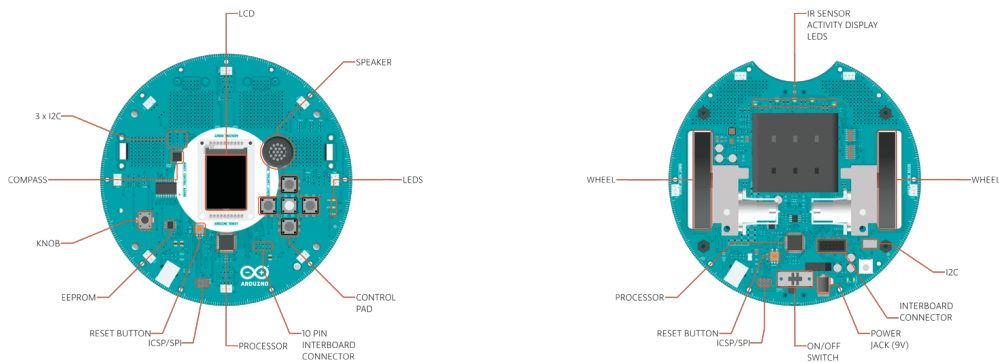
A unidade de controlo desta plataforma é um Microcontrolador de 32 bits com arquitetura *Acorn RISC Machine* (ARM), o Stellaris LM3S9B92 da Texas Instruments. Este Microcontrolador possui uma frequência máxima de funcionamento de 80 MHz, 256 KB de memória flash e 96 KB de memória SRAM.

2.5.5 Arduino Robot

O *Arduino robot* é uma plataforma robótica que enriquece a oferta do mundo Arduino. É um robô com uma estrutura composta por duas PCBs que se encaixam uma no topo da outra. Tem uma forma circular com 19 cm de diâmetro e está disponível no mercado por 189 EUR.

Uma das PCBs, a inferior, está mais dedicada à parte de potência: controlo de alimentações e dos motores. Já a PCB superior apresenta a unidade de controlo e também uma interface com o utilizador, baseada num display e 5 botões de interface.

Na Figura 2.11 pode-se observar as duas bases desta plataforma legendadas em inglês. Com as posições dos vários componentes na plataforma.



(a) Plataforma Arduino Robot, PCB superior [41] (b) Plataforma Arduino Robot, PCB inferior [41]

Figura 2.11: Plataforma Arduino Robot

Suporte sensorial

Esta plataforma apresenta sensores de superfície semelhantes aos que apresentava a plataforma Pololu 3 π , apresenta também uma bússola. Para além destes, existem também 5 botões e um potenciômetro.

Atuadores

Tal como nas anteriores, esta plataforma apresenta um sistema de locomoção diferencial. Assim, há dois motores de locomoção para atuar, para além disso, há ainda LEDs e um *display* a cores.

Suporte para processamento

Cada PCB contém a sua própria unidade de processamento baseada num Microcontrolador ATmega32u4, com arquitetura AVR de 8 bits, uma frequência máxima de funcionamento de 16 MHz, 32 KB de memória flash e 2,5 KB de memória RAM. Adicionalmente existe uma unidade EEPROM de 64 KB acessível através de I2C e suporte para a adição de um cartão de memória.

2.6 Conclusão

Tem-se assistido ao uso de robôs na educação nas mais variadas disciplinas e nos mais variados níveis de ensino. O objetivo é ensinar, contudo, a maneira como tal é conseguido nem sempre é clara. Sabemos que os alunos se divertem com a experiência e sabemos que o

uso de robôs promove um incremento da aprendizagem de competências como o projeto e o trabalho em grupo.

O DETI tem apostado em realizar várias iniciativas de robótica didática. Em algumas delas construíram-se plataformas robóticas educacionais que permitem aos participantes concentrarem-se em problemas do âmbito da iniciativa sem terem de dominar as áreas da eletrónica ou da mecânica necessárias para a construção destas plataformas. Apesar de participar com uma plataforma robótica, o participante pode querer alterá-la, pelo que é interessante que exista uma maneira fácil de incluir alterações.

Das plataformas que se referiu pode-se ver que as que foram desenvolvidas no DETI apresentam uma estrutura mecânica onde assentam todos os componentes elétricos que constituem a plataforma. Já nas plataformas referidas que existem no mercado, uma PCB implementa a parte eletrónica da plataforma e é, ao mesmo tempo, o suporte mecânico para todo o robô. Com esta dissertação pretende-se alargar esta ideia e construir um robô expansível com um maior conjunto de sensores que assenta na mesma ideia base de uma PCB ser o suporte para a maioria dos componentes mecânicos, sensoriais e de atuação.

Capítulo 3

Projeto e Construção da Plataforma

Como já foi discutido na introdução, ao longo dos anos em que se realizou o concurso MicroRato e para apoio a outras atividades de robótica educacional têm sido desenvolvidas várias plataformas robóticas educacionais no DETI. Já se sabe que tipo de sensores, atuadores e suporte de processamento possuem, por isto, neste momento, fazendo uma comparação entre estas plataformas e as que estão disponíveis no mercado e, tendo em mente os requisitos referidos na Introdução, pode-se fazer o levantamento das funcionalidades que a plataforma a construir deve apresentar. Contudo, antes disso ainda, deve-se fazer a conceptualização da plataforma em que se vai referir os desafios que se tem pela frente e deixar mais claro qual é a ideia em concreto.

3.1 Conceptualização da plataforma

Em relação às plataformas anteriormente desenvolvidas na Universidade de Aveiro, nesta, pretende-se reduzir o suporte mecânico a uma PCB que serve de estrutura da plataforma.

Existem vários requisitos de desenho que devem ser cumpridos para se conseguir obter os objetivos pretendidos. O conjunto de sensores e atuadores a introduzir deve ser semelhante ao das plataformas anteriores, uma vez que, a ideia base é construir uma plataforma que possa participar no concurso Micro-Rato. Pode-se, eventualmente, alargar o suporte sensorial e de atuação. Sabe-se que o tamanho de uma PCB influencia o seu preço pelo que é do nosso interesse reduzi-lo o máximo possível, assim deve-se ter cuidado na escolha do que vai figurar na plataforma e, ainda mais importante, deve-se ter cuidado na localização dos diversos componentes na PCB para que se consiga aproveitar melhor o espaço. Ao mesmo tempo que se reduz o tamanho e número de componentes ao mínimo necessário, está-se também a reduzir o peso da plataforma o que vai levar a que não necessitemos de motores de locomoção tão potentes, contribuindo novamente para a redução do peso e do tamanho da plataforma.

Um dos grandes desafios de construir uma plataforma como a esta é a sua estrutura monolítica. Nas plataformas anteriores, uma vez que a estrutura física não estava fortemente relacionada com a estrutura eletrónica, como neste caso em que coincidem, podia-se fazer um projeto mais modular, em que cada módulo ficaria no seu espaço. Assim, no futuro, havia a possibilidade de substituir um deles sem afetar os outros e erros num dos módulos não iam ser tão custosos no processo de projeto e *debug* da plataforma. Por outro lado esta abordagem, por necessitar de poucas peças mecânicas adicionais e de menos fios, torna mais fácil o processo de montagem. O teste elétrico da plataforma sai também facilitado por não haver nenhum

impedimento ao acesso à componente eletrônica. Contudo, esta exposição torna a plataforma menos segura uma vez que é mais fácil danificar a parte elétrica. Convém também lembrar que a robustez física desta solução é menor do que a das plataformas anteriores.

Tendo então todos estes desafios em mente, vamos idealizar a plataforma como uma PCB em que encaixam todos os componentes. Na Figura 3.1 podem-se ver duas vistas elucidativas do que se pretende. Em primeiro lugar nota-se uma forma octogonal da PCB, isto porque desta forma a sua realização fica mais fácil por oposição a uma forma redonda. A posição das rodas mostra um sistema de locomoção diferencial que já figurava em algumas das plataformas abordadas no capítulo anterior. A haste e o detetor de farol fazem parte do sistema de deteção do farol que poderá ser montado opcionalmente, contudo, o projeto deve ser feito a contar com ele.

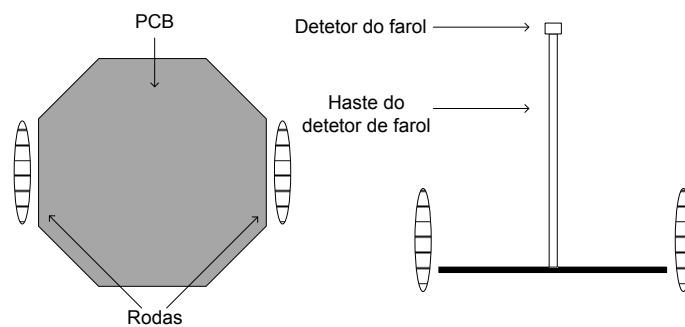


Figura 3.1: Conceito base da plataforma, à esquerda vista de topo, à direita de trás

O processo de conceptualização e projeto da plataforma, apresentado neste capítulo, visa descrever o trabalho desenvolvido no âmbito desta dissertação para a construção da plataforma robótica. Ir-se-á começar por descrever as funcionalidades que se querem presentes na plataforma e de seguida partir para a forma como se implementa cada uma delas. Por fim referir-se-á o *software* implementado de forma a criar uma camada de abstração da camada física do robô. Antes de prosseguir deve-se apenas referir alguns pormenores importantes sobre a forma como será escrito este capítulo. O texto será escrito em português contudo os termos técnicos estarão sempre na língua original. Sempre que se apresentar uma imagem da plataforma robótica a parte da frente estará virada para cima, considera-se a parte da frente aquela que fica orientada para o sentido em que o robô se movimenta quando se desloca com uma velocidade positiva.

3.2 Funcionalidades da plataforma

Para a apresentação das funcionalidades da plataforma, seguir-se-á a abordagem segundo a qual se apresentaram as plataformas robóticas no capítulo anterior, apresentando o suporte sensorial, atuadores e suporte de processamento. Adicionalmente, devido à importância que representa nesta plataforma vai-se também abordar a comunicação e a expansibilidade.

Por o projeto estar, em muito, relacionado com o concurso Micro-Rato, várias funcionalidades têm a sua inclusão motivada por ele, pelo que serão incluídas várias referências ao contexto em que este decorre.

Nesta fase está-se apenas a fazer um levantamento das funcionalidades, pelo que não interessa referir como será feita a sua implementação.

3.2.1 Suporte Sensorial

Deteção de obstáculos

Se se pensar no contexto do concurso Micro-Rato o robô é colocado num labirinto que desconhece *a priori* tendo de se orientar no mesmo [3]. Sendo assim, é imperativo que o robô seja capaz de detetar obstáculos no seu caminho para que os possa evitar.

Deteção de zonas não refletoras no chão

O ponto de chegada, é assinalado por uma zona não refletora no chão. Sendo assim o robô deve ter a possibilidade de a detetar. Em consequência, este será capaz de distinguir entre zonas refletoras e não refletoras. Pode-se usar esta funcionalidade, tendo o devido cuidado na sua implementação para, por exemplo, fazer o robô seguir uma linha refletora/não refletora num chão não refletor/refletor respetivamente.

Deteção de um farol de infra-vermelhos

No ponto de chegada existe um farol emissor de luz infravermelha para o qual o robô se deve orientar. A plataforma robótica será então dotada de um recetor adaptado para as características da radiação emitida pelo farol.

Medição da distância percorrida pelo robô

Ao incluir esta funcionalidade no robô existe uma forma de medir o seu deslocamento. Tal pode ser de extrema utilidade para o cumprimento do segundo objetivo, o regresso ao ponto de partida. Pode servir também para fechar a malha de controlo do sistema de locomoção e corrigir eventuais imperfeições dos motores de locomoção.

3.2.2 Atuação

Locomoção diferencial

O robô, sendo móvel, toma a sua mobilidade de um sistema de locomoção diferencial. Trata-se de um dos sistemas mais simples de locomoção, quer a nível de controlo, quer a nível de construção [42]. Quer-se, neste ponto, que o robô tenha uma velocidade tão elevada quanto o possível sem comprometer a sua performance em relação ao ambiente, no fundo pretende-se um robô rápido e ágil.

3.2.3 Comunicação

É importante que exista um modo de comunicação entre o robô e uma entidade externa. Em primeiro lugar, esta serve para que haja um modo de programação da unidade de controlo e *debug* do *software*. Depois, baseadas nesta comunicação, pode-se criar uma série de funcionalidades quer para um utilizador final quer para o criador de aplicações. A comunicação vem facultar o conhecimento do estado do robô, e a atuação no mesmo através da entidade externa.

Numa primeira instância a comunicação será cablada podendo esta manter-se para uso futuro. Contudo, a comunicação cablada conduz a inconvenientes que para um robô móvel são de certa forma óbvios, pois limitam a sua mobilidade. Assim surge a necessidade de haver uma comunicação sem fios. A comunicação sem fios vem permitir as mesmas funcionalidades da comunicação cablada sem comprometer a mobilidade do robô.

3.2.4 Suporte de Processamento

A conjugar tudo aquilo que foi falado até agora existirá uma unidade de processamento. O cuidado na escolha desta é muito importante para o uso futuro do robô. Não se quer que a plataforma se venha a tornar limitada por as suas capacidades de processamento serem demasiado baixas, mas também não se quer comprometer o consumo energético, o custo e a facilidade de desenvolvimento com uma unidade de controlo desproporcionada, em que não haja muita experiência ou com uma curva de aprendizagem demasiado longa.

3.2.5 Expansibilidade

A expansibilidade torna-se importante uma vez que se pretende que esta seja uma plataforma de suporte para futuras aplicações. Quando se refere a expansibilidade fala-se em construir o robô de forma a que seja fisicamente possível colocar na sua estrutura periféricos que podem vir a ter as mais variadas aplicações. É possível distinguir entre dois casos de expansão. No primeiro, o periférico a colocar é pequeno o suficiente para poder ser adaptado na placa de circuito impresso que constitui a base do robô, devendo esta ser dotada de áreas livres com suporte físico para expansão. No caso em que os periféricos são demasiado grandes e a opção de os colocar na base do robô é inviabilizada existirá um conjunto de furos que podem ser usados para segurar fisicamente o periférico. É importante que exista o suporte básico de comunicação e fornecimento de energia a esses periféricos.

3.3 Arquitetura da plataforma

Neste momento, sabe-se que se quer construir uma plataforma robótica educacional em que se irão colocar os seus vários constituintes numa PCB octogonal. Neste ponto do projeto, decide-se como se vai implementar as funcionalidades acima referidas e como integrar essas soluções de implementação na plataforma.

Vai-se então abordar, para cada funcionalidade referida, a sua implementação e localização na plataforma. O ponto de partida é a ideia já apresentada na Figura 3.1.

3.3.1 Suporte sensorial

Deteção de obstáculos

Para a deteção de obstáculos à frente do robô, várias técnicas podem ser empreendidas. Nas plataformas robóticas anteriores foram usados sensores Sharp GP2D12 que permitem detetar obstáculos em distâncias que podem variar entre 10 e 80 cm [43]. Atualmente estes sensores foram descontinuados pela Sharp e substituídos por uma versão mais recente: o GP2Y0A21 [44].

Na Tabela 3.1 pode-se ver uma análise comparativa entre várias técnicas e soluções de deteção de obstáculos. Usou-se como referência [8] e extraiu-se a informação referente a alguns

sensores de ultrassons e sensores de infravermelhos que funcionam com base em triangulação. Os sensores que funcionam com base em triangulação, usam um método que faz depender a sua saída do ângulo de incidência da radiação refletida no recetor de radiação infravermelha.

O preço é um componente que deve ser analisado, pois interessa construir uma plataforma de baixo custo.

Tabela 3.1: Comparação entre várias alternativas para medição de distâncias. Fonte: [8]

Técnica de detecção	Sensor	Alcance(m)	Tensão(V)	Preço
Ultra sons	LV-MaxSonar-EZ3	0,15 a 6,45	2,5 a 5	26,95USD
	DFRobot URM	0,04 a 5	5	16,85USD
Triangulação	GP2Y0A21	0,1 a 0,8	5	11,75USD
	DMS-80	0,1 a 0,8	5	14,90USD

Como é possível concluir da análise da tabela, os sensores que funcionam com base em triangulação apresentam distâncias de detecção adequadas à aplicação em questão. Optou-se pela utilização de sensores de infravermelhos com base em triangulação por serem um bom compromisso custo-funcionalidade.

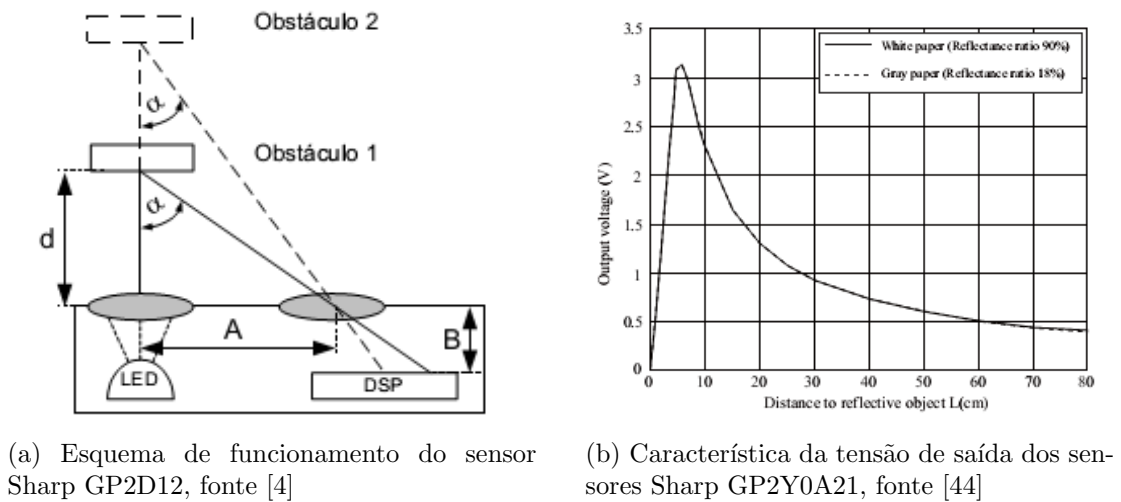


Figura 3.2: Característica da saída e funcionamento dos sensores Sharp

Para detetar a distância a que se encontra um obstáculo, estes sensores usam um método que é descrito na Figura 3.2a. A radiação refletida no obstáculo incide no recetor com um certo ângulo que é maior quanto maior for a distância ao obstáculo (notar que este ângulo é igual a $90^\circ - \alpha$). O recetor baseado num *Position Sensitive Detector* (PSD), vai produzir uma tensão de saída, V_o que é diretamente proporcional à distância do ponto de incidência da radiação recebida ao lado do PSD mais próximo do emissor. Assim a distância do ponto de incidência do feixe refletido no PSD ao emissor, é inversamente proporcional à distância d , consequentemente também V_o é inversamente proporcional à distância d . Tendo em conta também A e B concluímos que $V_o \propto (A \times B)/d$. Pode-se comprovar esta relação na Figura 3.2b.

Na versão descontinuada destes sensores, GP2D12, a curva característica revelava um comportamento diferente da apresentada na Figura 3.2b. É possível ver que a curva apresenta

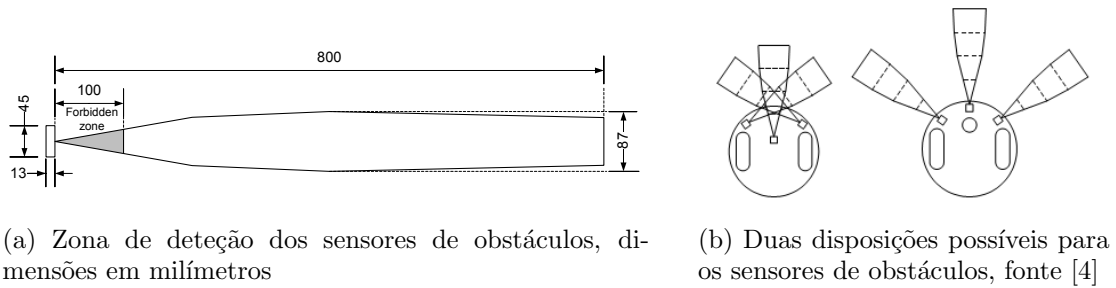


Figura 3.3: Zona de detecção e várias disposições dos sensores Sharp GP2D12

um ponto de inflexão para uma distância de aproximadamente 5 cm, se um obstáculo se aproximar mais do que essa distância vai, do ponto de vista da saída do sensor, parecer que se está a afastar. No caso do sensor GP2D12, esse ponto de inflexão ocorre para a distância de 10 cm, daí ter-se afastado os sensores essa distância dos limites da plataforma em plataformas anteriores. Neste caso, usando os sensores GP2Y0A21 necessita-se apenas de os afastar cerca de 5 cm.

Estes sensores têm uma zona de detecção limitada dentro da qual deve estar um obstáculo de forma a ser detetado. A zona de detecção pode ser vista na Figura 3.3a. Na Figura 3.3b representa-se a sobreposição das várias zonas de detecção individuais de três sensores colocados na parte frontal de uma plataforma em duas disposições diferentes. Deve-se ter presente que a disposição dos sensores é muito importante para a performance do robô, pois define a forma como este se comporta na presença de um obstáculo. Na disposição à direita da Figura 3.3b não há sobreposições entre as zonas de detecção dos sensores, contudo, tal disposição cria uma série de ângulos mortos imediatamente à frente do robô. Tem-se optado pela disposição esquematizada à esquerda da Figura 3.3b, o que, para além de reduzir os ângulos mortos imediatamente à frente do robô, leva a que um obstáculo nessa zona possa ser detetado por todos os três sensores ao mesmo tempo.

Anteriormente, tinha-se optado por colocar os sensores laterais num ângulo de 120° com o plano de detecção do sensor central, o que levava a que, para que o ponto de interseção das três áreas de detecção fosse próximo do robô, se devessem afastar os sensores tal como se pode ver na Figura 3.4a. Como neste caso se pretende instalar os sensores no mínimo espaço possível optou-se por aumentar o ângulo referido para 135° e colocar os sensores mais próximos. A forma escolhida pode ser vista na Figura 3.4b onde se pode ver que esta solução leva a que os sensores ocupem um espaço mais compacto. No Anexo 4, é possível ver o robô com as áreas de detecção resultantes em maior detalhe.

A interface com o sensor faz-se por meio de uma saída analógica que este apresenta com as características de tensão da curva já referida. O esquema elétrico da ligação destes sensores pode ser encontrado no Anexo 1.

Uma característica deste sensor que é determinante para o controlo global da plataforma é o tempo de processamento de uma nova medida de distância que é, em média, 38,3 ms [44]. Por este ser o sensor com uma atualização de dados mais lenta, como se irá ver nos sensores subsequentes, todo o controlo é determinado por esta característica temporal.

Do que foi referido, resulta a introdução de três sensores de obstáculos de modo a poder detetar obstáculos à frente do robô. Olhando para a sua zona de detecção nota-se que há uma distância mínima a que estes sensores devem ser colocados do chão, pois se este entrar na sua

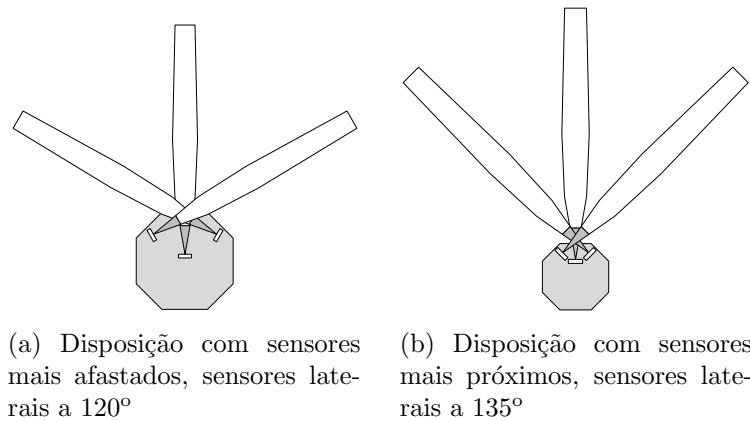


Figura 3.4: Duas disposições em ângulos diferentes dos sensores de obstáculos

zona de deteção vai ser identificado como um obstáculo. Assim, consultando o anexo Anexo 1, é possível concluir que estes sensores não podem ser colocados a uma distância inferior a 43,5 mm do chão. Sabe-se também que não se pode colocar nada à frente dos sensores que obstrua a visão para os obstáculos.

O esquema conceptual da localização dos sensores resultante pode ser visto na Figura 3.5. Para a fixação destes na plataforma robótica foi projetada uma PCB que pode ser encontrada no Anexo 3.

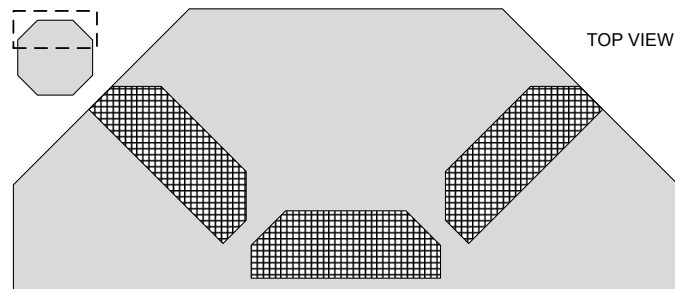


Figura 3.5: Disposição dos sensores de obstáculos no robô, assinalados a tracejado preto, vista de topo.

Deteção de zonas não refletoras no chão

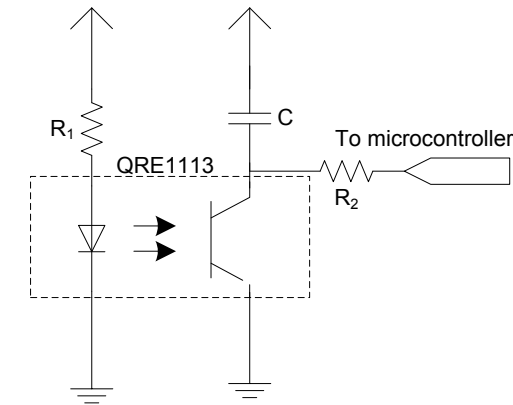
A deteção de zonas não refletoras no chão tem sido feita por meio de um par emissor-refletor de luz infravermelha. Quando o chão não é refletor, o feixe emitido é interrompido e não chega ao recetor, é então possível detetar entre um chão refletor e um chão não refletor. Adicionalmente, pode-se combinar os sensores de forma a que seja possível o seguimento de uma linha.

Os sensores escolhidos foram os QRE1113 [45]. Estes sensores são de reduzidas dimensões e, como incluem no mesmo encapsulamento o recetor e o emissor de radiação infravermelha, torna-se fácil a sua inclusão na plataforma.

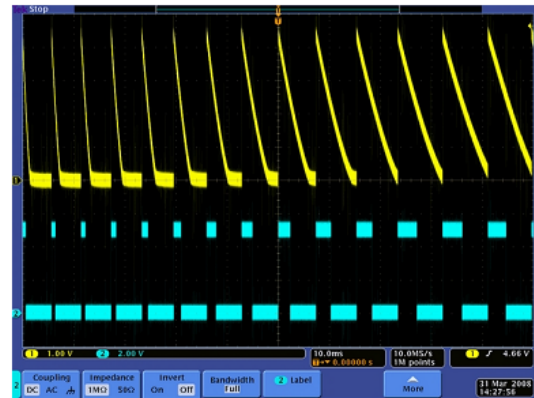
Na Figura 3.6a pode-se ver o circuito elétrico de ligação destes sensores. No Anexo 1 encontra-se o circuito elétrico para todos os sensores. Procedendo à análise da montagem

referida, do lado do emissor esta é fácil de entender, apenas se necessita de uma resistência em série para polarizar o emissor de luz infravermelha. Contudo, do lado do recetor, poderia afirmar-se que uma resistência ligada entre a fonte de alimentação e o coletor do fototransistor seria suficiente. Nesse caso, este estaria cortado se o chão fosse completamente não refletor. Caso o chão fosse refletor o fototransistor estaria saturado. A esse estado de aberto e fechado corresponderiam níveis de tensão que se queria que fossem próximos da tensão de alimentação, caso do chão não refletor, e próximos de zero, caso do chão refletor. Como já se disse que esta característica de chão completamente refletor ou não refletor não se verifica na realidade, esta montagem com uma resistência entre a alimentação e o coletor do fototransistor exigiria uma cuidada escolha do valor da resistência para que se verificassem os níveis de tensão desejados. Afim de contornar esse problema, optou-se por usar uma montagem com o condensador C e a resistência R_2 como se pode ver na Figura 3.6a.

Os valores destes componentes foram escolhidos seguindo a recomendação da *application note* referida em [46] e foram 10 nF para C e 220 Ω para R_2 .



(a) Circuito de suporte para um sensor QRE1113



(b) Característica da tensão na saída do circuito de suporte para o sensor à medida que uma superfície não refletora entra no campo de deteção do sensor. A amarelo, o sinal analógico presente no porto do microcontrolador. A azul, leitura digital desse valor. fonte [46]

Figura 3.6: Circuito de suporte para o sensor QRE1113 e sua saída característica

O procedimento de leitura, que deve ser usado, pode ser descrito nos seguintes passos:

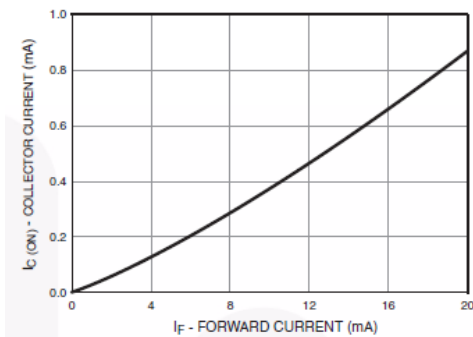
1. Configurar o porto do microcontrolador como saída e colocá-lo no nível lógico '1' para descarregar o condensador C ;
2. Após aguardar que o condensador descarregue, configura-se o porto do microcontrolador como entrada e mede-se o tempo que, com a descarga do condensador C , o valor do nível lógico lido passe a ser '0'.

Se se respeitar os passos acima descritos pode-se definir dinamicamente os tempos que se atribuem a um chão não refletor e a outro refletor, sendo que se pode definir um limiar de

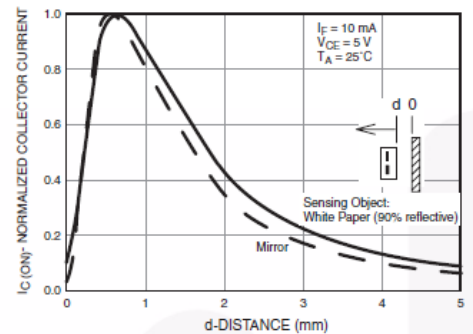
decisão abaixo do qual se considera que o chão é refletor e acima do qual se considera não refletor. Para este valor de limiar de decisão demonstrou-se, praticamente, ser apropriado o valor de 7ms para um par de materiais preto e branco. Contudo, pode-se ajustar, por software, o limiar de decisão para outro par de materiais, pode até introduzir-se histerese na decisão.

O conjunto de passos definido leva às formas de onda apresentadas na Figura 3.6b, medidas quando uma superfície não refletora entra no campo de detecção do sensor onde é possível notar o aumento do tempo de carga do condensador, em resultado de uma corrente menor no fototransistor.

A inclusão destes sensores limita a distância a que a plataforma deve estar do chão: se esta for grande demais a potência da radiação refletida pode ser tão atenuada que o condensador demore demasiado tempo a carregar. Uma solução para minimizar este problema passa por aumentar a corrente do díodo emissor de radiação infravermelha, conseguindo, assim, que a quantidade de radiação recebida seja maior. Existe, então, um compromisso entre a energia gasta e a distância ao chão. Para ajudar na decisão da distância mínima ao chão podem-se consultar o *datasheet* deste sensor onde se encontra o gráfico da Figura 3.7b que mostra como evolui a corrente no coletor do fototransistor que constitui o recetor de infravermelhos. Pode ver-se que para distâncias superiores a 5 mm a corrente no fototransistor cai para valores demasiado baixos que levariam a um aumento do tempo de carga do condensador C . Contudo aumentando a corrente de polarização do emissor de infravermelhos pode-se aumentar este valor, como se vê no gráfico da Figura 3.7a. Assim, decidiu-se colocar estes sensores a aproximadamente 7 mm do chão e polarizou-se os emissores de radiação infravermelha com 30 mA, sendo que o valor máximo permitido é de 50 mA.



(a) Variação da corrente de coletor do fototransistor com a corrente de polarização do emissor de infravermelhos [45]



(b) Variação da corrente de coletor com a distância a um obstáculo [45]

Figura 3.7: Caraterização elétrica do comportamento do sensor QRE1113

A posição decidida para estes sensores pode ser vista na Figura 3.8. Um desenho com mais rigor pode ser encontrado no Anexo 4. Optou-se por posicionar os sensores na frente da plataforma pois, para aplicações de seguimento de linha faz mais sentido. Os 3 sensores centrais são suficientes para fazer o seguimento de uma linha, contudo, decidiu-se incluir mais 2, pois assim podem também detetar-se interseções de linhas o que pode ser útil para resolver labirintos, por exemplo.

Estes sensores poderão ser chamados sensores de superfície, de chão ou de linha no texto desta dissertação.

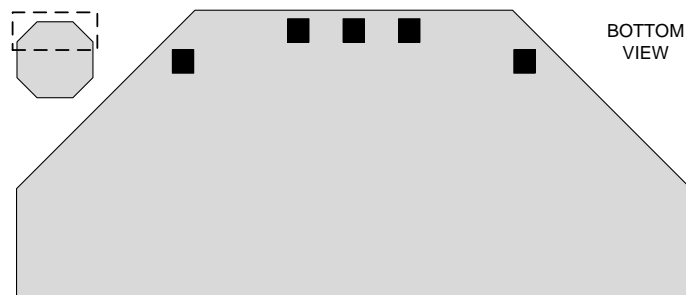


Figura 3.8: Disposição dos sensores de superfície no robô, assinalados a preto, vista de baixo.

Deteção de um farol de infravermelhos

No concurso Micro-Rato o ponto de chegada é assinalado por meio um farol de infravermelhos para o qual os robôs participantes se devem orientar. A radiação infravermelha emitida por este farol tem características específicas. Tem um comprimento de onda de 940 nm, o sinal é modulado usando uma modulação ASK com um sinal periódico de 600 Hz com um *duty-cycle* de 30 % sobre uma onda de 38 kHz [3]. O sinal resultante pode ser visto na Figura 3.9.

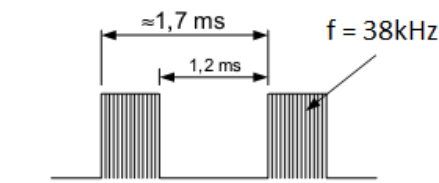


Figura 3.9: Sinal emitido pelo farol de infravermelhos usado na competição Micro-Rato.

O recetor tem de ser apropriado para as características referidas. Para receber este tipo de sinal podem ser usados os circuitos normalmente utilizados nos receptores de telecomando das televisões, uma vez que se baseiam no mesmo tipo de modulação e usam a mesma frequência de portadora. Assim, há facilidade em encontrar no mercado sensores que podem ser usados nesta aplicação.

Pode usar-se, por exemplo, o sensor GP1UE281YK da Sharp que faz a desmodulação do sinal e o converte na modulante (uma onda quadrada com uma frequência de 600 Hz, neste caso) [47]. Na aplicação destes sensores deve-se ter em conta que, em primeiro lugar, se deve discriminar com alguma certeza a direção em que se encontra o farol e, em segundo lugar, se tem de fazer a interface entre o sinal de saída do sensor e a unidade de controlo.

Para saber em que direção se encontra o farol deve-se colocar o recetor escolhido no mesmo plano que os emissores do farol, assim, este deve estar a 28 cm do chão. Para saber a direção com alguma precisão decidiu-se optar pela abordagem que foi usada nas plataformas anteriores: colocar o sensor na parte traseira de um tubo comprido fazendo com que a deteção se dê com mais precisão e acoplar este conjunto a um servo-motor capaz de rodar 180° . Saber em que direção está o servo-motor quando o farol está em linha de vista com o robô, permite saber em que direção se encontra a posição do farol em relação à direção da frente do robô. Esta abordagem conduz à necessidade de prever na base do robô uma forma de adaptar toda a estrutura necessária para apoio a este sensor. Nas Figuras 3.10 e 3.11 pode-se ver a solução

completa quer para a PCB do robô numa vista de topo, quer para a estrutura de suporte ao sensor de farol numa vista lateral. O sensor de farol é colocado na parte traseira do número 4 na Figura 3.11.

Este conjunto pode ser montado opcionalmente no robô, uma vez que as suas aplicações não se restringem apenas ao concurso Micro-Rato em que o farol de infravermelhos é usado. Na altura da escrita desta dissertação este sensor não se encontrava montado na plataforma.

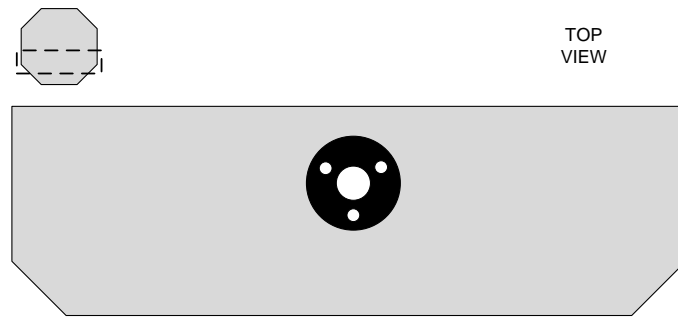


Figura 3.10: Posição do encaixe para o suporte do sensor.

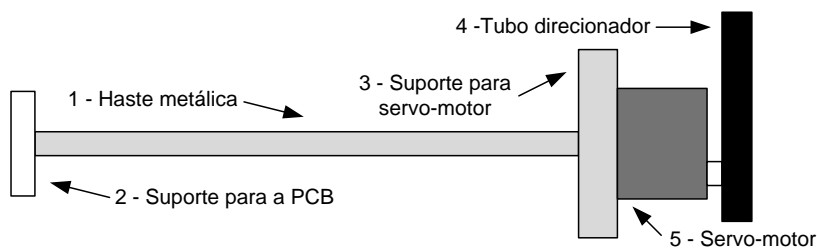
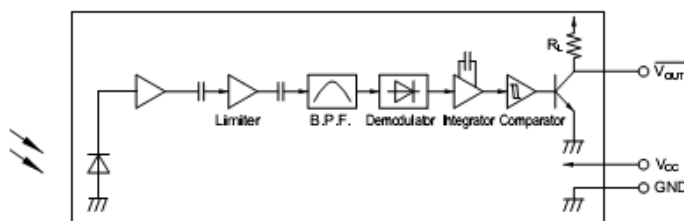
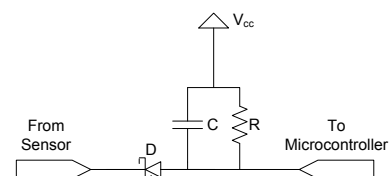


Figura 3.11: Estrutura necessária para o suporte do sensor.

Quanto à adaptação elétrica, esta deve-se às características que apresenta a saída do sensor de deteção do farol. Pode-se ver o esquema interno deste sensor na Figura 3.12a. É possível concluir da forma de onda apresentada anteriormente e deste esquema interno que a saída deste sensor vai ser o sinal modulante invertido, ou seja, uma onda quadrada com uma frequência de 600 Hz e um *duty-cycle* de 70 %. Um microcontrolador a detetar esta envolvente envolveria algum processamento adicional que pode ser poupado usando a montagem apresentada na Figura 3.12b, que é um detetor de envolvente.



(a) Diagrama de blocos interno do sensor GP1UE28YK.



(b) Circuito detetor de envolvente

Figura 3.12

O funcionamento do detetor de envolvente pode ser explicado da seguinte forma: quando a tensão de saída do sensor é 0 V o diodo D conduz carregando o condensador C o que leva a tensão de saída para o microcontrolador a descer; quando a tensão de saída é próxima da tensão de alimentação do sensor (entre 2,4 e 3,6 V), o diodo mantém-se em corte, o que leva a que o condensador descarregue através da resistência R com uma constante de tempo menor que aquela com que foi carregado. Desta forma, quando o farol está em linha de vista, a tensão de saída do conjunto sensor mais detetor de envolvente mantém-se próxima de 0V, caso contrário esta mantém-se próxima da tensão de alimentação do conjunto.

Os valores de C e R foram dimensionados de forma a que o circuito não apresente uma queda de tensão significativa no tempo em que o condensador descarrega através da resistência R e com recurso aos circuitos semelhantes implementados em plataformas anteriores, [39]. Escolheu-se usar um condensador de 470 nF e uma resistência de 56 k Ω . Dos valores destes componentes é possível concluir que a constante de tempo do conjunto é $\tau = R \times C = 56 \text{ k} \times 470 \text{ n} = 26 \text{ ms}$. Tem-se, então, uma queda de 4,5 % nos 1,2 ms em que o condensador descarrega através de R . Um sobredimensionamento destes componentes leva a que, após o farol deixar de estar em linha de vista, se demore algum tempo a sabê-lo. Um subdimensionamento leva a que a subida de tensão que se dá quando o farol está em linha de vista e a onda quadrada se encontra a 3,3 V possa ser grande demais para que o microcontrolador assuma que o farol não está em linha de vista.

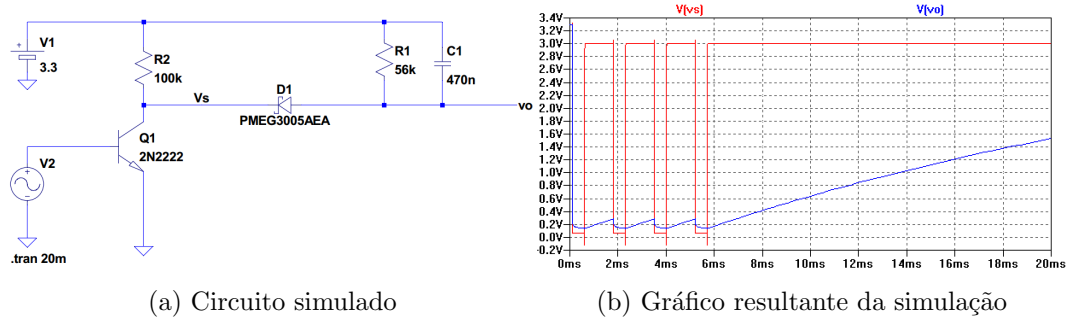


Figura 3.13: Circuito simulado e gráfico resultante da simulação

Foi feita uma simulação usando o circuito da Figura 3.13a e o programa de simulação LTspice, que permitiu validá-lo e caracterizá-lo. Teve-se em conta a saída do sensor de farol, incluindo na simulação o transistor $Q1$ e a resistência $R2$. Na Figura 3.13b podem-se ver o gráfico resultante desta simulação onde estão representadas as curvas $V(vs)$ correspondente à tensão de saída do sensor e a curva $V(vo)$ correspondente à saída do circuito implementado. Como se pode ver, quando o farol está em linha de vista, ou seja a saída do sensor apresenta uma onda quadrada com uma frequência de 600 Hz e *duty-cycle* de 70 %, a saída vo mantém uma tensão nunca superior a 0,3 V, quando o farol não está em linha de vista a tensão nesta saída tende para a tensão de alimentação do circuito que é, neste caso, 3,3 V. O valor de V_{IL} do circuito de controlo digital a que este conjunto for ligado deve ser superior a 0,3 V, caso não o seja deverão ser feitas alterações nos valores de $C1$ e $R1$ de forma a aumentar a constante de tempo de descarga do condensador $C1$.

Medição da distância percorrida pelo robô

Para a medição do deslocamento do robô podem empreender-se várias técnicas. Uma vez que se irá usar um método de locomoção diferencial, como já foi anunciado atrás, uma das formas mais acessíveis de fazer a medição da distância percorrida pelo robô é acoplar *encoders* aos motores de locomoção. Estes dispositivos permitem medir, diretamente ou não, a distância percorrida pela roda movida pelo motor.

Escolheu-se para esta aplicação o *encoder* AEAT-6012 da *Avago Technologies*. Este *encoder* mede a posição absoluta de um íman posicionado sobre o circuito integrado em que está implementado. A versão referida obtém a posição com 12 bits de resolução, ou seja, para uma volta completa, valores entre 0 e 4095. É possível ver na Figura 3.14, à direita, o íman assinalado como *Magnet* que o veio do motor faz rodar, solidariamente consigo, muito próximo da PCB assinalada como *Sensor PCB assembly*. Nesta, está o circuito integrado que implementa o *encoder*.

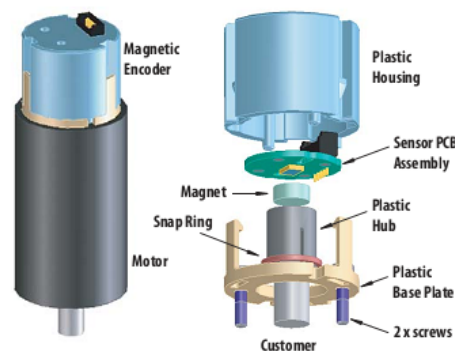


Figura 3.14: À esquerda: *encoder* usado montado num motor. À direita: Esquema interno do *encoder* [6].

Este *encoder* amostra periodicamente a posição do íman com uma frequência entre 9,9 e 10,94 kHz, rápido o suficiente para a nossa aplicação. A ligação ao exterior é efetuada através de uma ligação *Synchronous Serial Interface* (SSI), que é muito similar ao protocolo *Serial Peripheral Interface* (SPI) [6]. À semelhança do SPI está disponível uma linha de *clock*, uma saída de dados série e um *chip select* que permite controlar o acesso à linha onde estarão ligadas as saídas dos *encoders*, que serão 2 neste caso. O funcionamento da comunicação SSI é melhor entendido com o auxílio da Figura 3.15, onde se vê um diagrama temporal de uma leitura de um destes *encoders* na sua versão de 10 bits.

A este esquema estão associadas características temporais que podem ser consultadas na Tabela 5.1, no Anexo 5.

Necessita-se de determinar a necessidade que há em termos de frequência de amostragem dos dados dos *encoders*. Tratando-se de um *encoder* absoluto, para que se possa discriminar o sentido de rotação do veio do motor, a frequência de leitura dos dados dos *encoders*, deve ser maior que o dobro da velocidade de rotação máxima. Torna-se necessário adiantar alguns parâmetros do sistema de locomoção do qual se falará mais à frente. Assim, resume-se na Tabela 3.2 alguns dados necessários sobre os motores usados. É de realçar que os motores pelos quais se optou têm disponível na parte traseira uma extensão do veio principal à qual será possível acoplar o *encoder*.

Ao motor está acoplada uma caixa de redução, com um fator de desmultiplicação de 30,

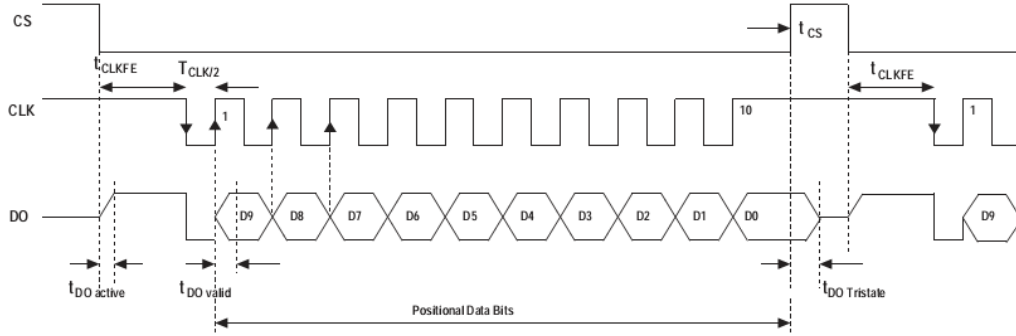


Figura 3.15: Diagrama temporal de uma operação de leitura do *encoder*. [6]

Tabela 3.2: Parâmetros do sistema de locomoção. Fonte: [9]

Parâmetro	Valor
Rotações por minuto máximas do veio de saída(v_{rpm})	291 rpm
Redução da caixa redutora r_{caixa}	30

o que se traduz, para o motor usado, numa velocidade de rotação máxima, no veio primário, de 8730 rpm, ou seja, 145,5 Rot/s. Assim, conclui-se que se deve amostrar a posição do veio do motor a uma taxa maior do que 291 Hz.

Com esta configuração, o número de pontos por volta, no veio de saída, é dado pela expressão seguinte, de onde se pode inferir que se obtém uma resolução angular de, aproximadamente, 0,0029°.

$$N_{ppr} = 2^{N_{bits}} \times r_{caixa}$$

$$N_{ppr} = 2^{12} \times 30 = 122880$$

Para o caso prático das leituras destes *encoders* os valores utilizados em termos de parâmetros temporais e número de bits foram condicionados pela implementação em *software* realizada. Assim resume-se na Tabela 3.3 os parâmetros usados.

Tabela 3.3: Resumo de parâmetros usados para obtenção de dados dos *encoders*

Parâmetro	Valor
Frequência de amostragem	299,6 Hz
f_{CLK}	≈ 78 kHz
N_{bits}	5

Com esta implementação consegue-se, ainda assim, uma resolução angular de 0,375°, ou seja, 960 pontos por volta no veio de saída.

Uma vez que se está interessado em saber a distância percorrida e não a posição em que o veio do motor se encontra deve-se transformar a informação proveniente dos *encoders*, a

posição do veio do motor, na distância percorrida. Com o valor atual lido do *encoder* e o valor anterior pode-se obter a distância percorrida no intervalo de tempo entre as duas leituras.

Necessita-se de um modo de calcular a distância percorrida pelo veio do motor a partir das leituras sucessivas das suas posições angulares, o que se traduz num sistema cuja entrada é a nova posição obtida e que necessita da posição anterior para obter a distância percorrida. A Expressão 3.1 permite obter o deslocamento do veio do motor a partir da posição atual e da anterior. Uma vez que se garante que são lidos pelo menos dois pontos por volta já se sabe que a distância percorrida tem de ser, em valor absoluto, inferior a metade do número de pontos por volta. Se ao subtrair o valor anterior ao atual se obtiver um resultado superior, em valor absoluto, a metade de uma volta, tem de se proceder à correção do valor usando a expressão já referida.

$$D = \begin{cases} (Pos - Pos_{ant}) + 2^{N_{bits}}, & \text{se } Pos - Pos_{ant} < -a \\ Pos - Pos_{ant}, & \text{se } -a \leq Pos - Pos_{ant} \leq a \\ (Pos - Pos_{ant}) - 2^{N_{bits}}, & \text{se } Pos - Pos_{ant} > a \end{cases} \quad (3.1)$$

O valor da constante a é dado pelo valor inteiro da expressão seguinte:

$$a = \frac{2^{N_{bits}}}{2}$$

Como exemplo prático da aplicação da Expressão 3.1 pode-se recorrer à Figura 3.16 onde se vê ilustrado, num esquema circular, a distribuição das posições lidas pelo *encoder* no caso de leituras de 4 bits.

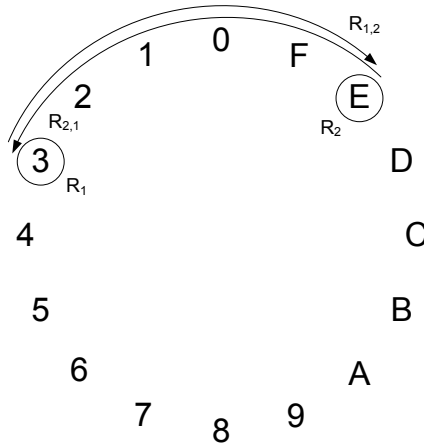


Figura 3.16: Valores lidos do *encoder* com leituras de 4 bits, valores em base hexadecimal

Vêm-se assinalados na Figura 3.16 os valores R_1 e R_2 obtidos de leituras consecutivas dos *encoders*. No caso em que R_1 é o valor atual e R_2 o anterior, a subtração destes dois valores dá $-0xB$ menor que a que neste caso é $0x7$ o resultado é, então, $0x8$. No caso em que R_2 é o valor atual e R_1 o anterior, a subtração dá $0xB$. Recorrendo à Expressão 3.1, chegamos ao valor $-0x8$.

Botões

Para servir de interface com o robô incluíram-se 3 botões cujas funções podem ser programada na unidade de controlo da plataforma. Estes botões ficaram localizados na parte traseira do robô como se pode ver na Figura 3.17. Adicionalmente foi reservado espaço para outro botão que serve para fazer o *reset* da unidade de controlo assinalado na Figura já referida como *Reset*.

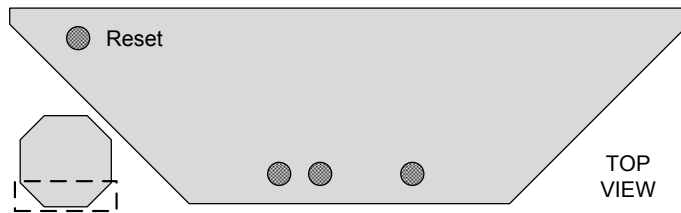


Figura 3.17: Posição dos botões na plataforma, assinalados a preto numa vista de topo.

3.3.2 Atuação

Locomoção diferencial

O esquema de locomoção implementado baseia-se em dois motores colocados como indica a Figura 3.18, em posições diametralmente opostas. Tendo em conta a diferença de velocidades entre as duas rodas é possível definir a trajetória a seguir pelo robô. O tamanho da plataforma robótica foi, em muito, determinado pelo tamanho dos motores. Dada a sua configuração é imposto um limite mínimo ao tamanho do robô. A distância entre as duas rodas na plataforma é de 184 mm como se pode ver na Figura 3.18.

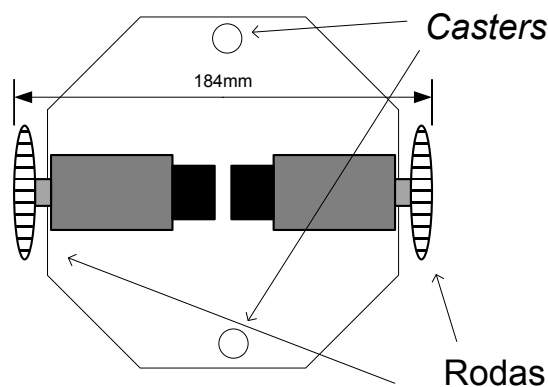


Figura 3.18: Modelo de locomoção diferencial com dois *casters*

Os motores pelos quais se optou foram os GHM-03 da Lynxmotion [9]. São motores DC de ímã permanente que apresentam uma velocidade máxima de 291 rpm o que, aliado a um diâmetro de rodas de 8 cm, permite obter uma velocidade máxima teórica de 1,2 m/s. Estes motores apresentam um conjunto de características que são resumidas na Tabela 3.2.

O acoplamento dos motores à plataforma é feito através de dois furos que foram deixados na PCB que constitui a base do robô com esse propósito. Esses furos fazem parte do *footprint*

Tabela 3.4: Características principais dos motores GHM-03. Fonte: [9]

Característica	Valor
Tensão nominal	7,2 V
Carga Nominal	810 g cm
Velocidade com carga nominal	235 rpm
Corrente com carga nominal	< 687 mA
Corrente máxima, carga de 3 kg cm	≈ 2 A

dos motores colocado na PCB. Mais pormenores sobre as características e posições destes furos podem ser encontrados no Anexo 4.

Para fazer o *drive* destes motores optou-se por uma ponte-H, em específico a TB6612FNG da Toshiba [48]. Esta é implementada num circuito integrado capaz de controlar dois motores simultaneamente com uma corrente média máxima de 1,2 A por motor. Será suficiente para fazer o drive dos motores de acordo com a Tabela 3.4 se não se exceder a carga nominal. O princípio de funcionamento da ponte-H pode ser melhor entendido pela observação das Figuras 3.19a e 3.19b. Fechando os *switches* assinalados como *S1* e *S4* na Figura 3.19a o motor roda num sentido que pode ser invertido, abrindo-os e fechando *S2* e *S3*. Afim de controlar a velocidade do motor, usa-se Pulse Width Modulation (PWM) como se vê na Figura 3.19b. Consegue-se, desta forma, ter controlo sobre a velocidade e a direção da rotação dos motores. Usou-se um sinal PWM com uma frequência de 20 kHz.

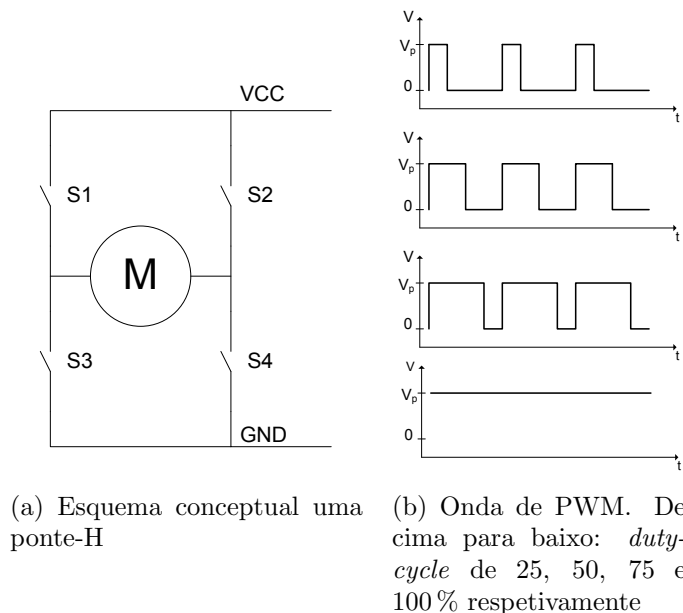


Figura 3.19: Funcionamento de uma Ponte-H

Um dos problemas deste sistema de locomoção é o *tilt* do robô, ou seja a sua inclinação para a frente numa travagem brusca ou para trás num arranque brusco. Para evitar o problema do *tilt*, pode-se usar um ou mais *casters*. Usando dois, como é dado a ver na Figura 3.18, tem-se a vantagem de resolver este problema de forma definitiva contudo cria-se um novo problema:

o robô pode perder o contacto com o chão numa superfície não plana. Um projeto cuidadoso permite atenuar estes problemas conjugando hardware e software. A nível de hardware opta-se por usar apenas um *caster*, o anterior, eliminado o posterior e deslocar o centro de massa para trás do eixo das rodas, assim em movimento uniforme o robô tenderá a manter-se estável. Contudo é necessário ter cuidado com a travagem para que não aconteça que o robô se incline para a frente, este cuidado deve refletir-se no software de controlo, criando uma travagem suave.

A introdução de *encoders* na plataforma veio permitir fazer um controlo dos motores de locomoção em malha fechada, o que mitiga imperfeições nos motores. O controlo foi implementado por meio de um esquema de compensação PI proporcional-integrador. O esquema de compensação baseia-se no diagrama apresentado na Figura 3.20. A introdução do compensador PI permite corrigir o erro em regime estacionário que estaria presente no caso de um compensador proporcional apenas. A parte integral permite também acelerar a resposta a variações de velocidade [49]. Ensaios práticos mostraram que 3 é um valor adequado para K_p e K_i .

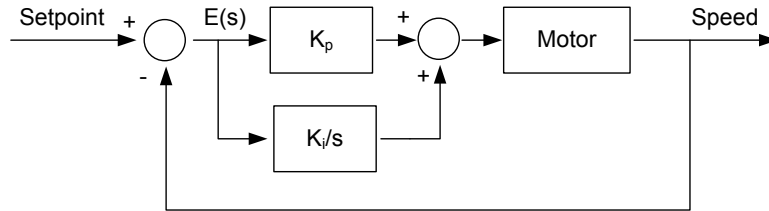


Figura 3.20: Esquema do compensador usado

As rotinas que efetuam a compensação são executadas no robô com uma frequência de 99,9 Hz que é 1/3 da frequência de amostragem dos *encoders*. Com esta frequência de controlo, obtém-se um máximo de deslocamento por volta, entre execuções das rotinas de compensação, de 45 pontos, este valor foi obtido a partir da Expressão 3.2 aplicada para o caso da velocidade máxima teórica que é de 1,2 m/s.

$$Set = \frac{\frac{v}{\pi \times d} \times N_{ppt}}{f_{ctr}} \quad (3.2)$$

Na Expressão 3.2 v é a velocidade da plataforma o que, dividido pelo perímetro da roda ($\pi \times d$ onde d é o diâmetro), dá o número de voltas por segundo que a roda efetua. Se se multiplicar este valor pelo número de pontos que o *encoder* lê por cada volta da roda, N_{ppt} , obtém-se o número de pontos por segundo. Por fim dividindo este valor pela frequência de controlo f_{ctr} obtém-se o número de pontos lidos entre execuções sucessivas das rotinas de compensação Set .

Servo-motor

A deteção do farol de infravermelhos já foi apresentada na secção 3.3.1, onde foi referida a presença de um farol de infravermelhos. Foi também referido que o detetor de infravermelhos seria movido para diferentes posições por meio de um servo-motor.

Este tipo de motores usa um mecanismo de realimentação para manter uma posição fixa em vez de rodar continuamente [50]. Um tipo específico de servos denominado *RC Servo* tornou-se muito popular na robótica e em aplicações controladas por rádio devido ao seu baixo peso, baixo preço e reduzidas dimensões. Este tipo de atuadores incorpora um pequeno motor DC uma caixa redutora e um potenciômetro como mecanismo de realimentação. A posição do potenciômetro é constantemente comparada com a posição do motor e é aplicada a este a tensão necessária para que se mantenha na posição pretendida. Vê-se na Figura 3.21 um destes atuadores aberto para que se possam observar os seus elementos constituintes.

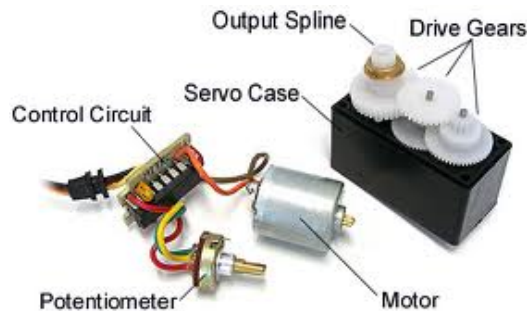


Figura 3.21: Constituintes de um servo-motor. Fonte: [7]

O controlo destes motores faz-se por meio de um sinal PWM em que a duração do pulso define a posição do servo-motor. A interface é tipicamente feita por meio de três condutores em que dois deles são usados para alimentar o servomotor com uma tensão contínua e o outro é usado para controlar a posição através de PWM.

O atuador escolhido foi o HS-5055MG da Hitec [10] que é um pequeno servo-motor com as características que são resumidas na Tabela 3.5. Para este servo-motor rodar 180° deve-se variar a duração do pulso da onda PWM de controlo entre 0,9 ms e 2,1 ms sendo que 1,5 ms define a posição central.

Tabela 3.5: Características principais do servo-motor HS-5055MG. Fonte: [10]

Característica	Valor
Tensão nominal	4, 8 a 6 V
Torque(4, 8/6 V)	1, 2/1,5 kg cm
Dimensões	$22,61 \times 11,43 \times 23,88 \text{ mm}^3$
Massa	9,36 g
Ângulo de rotação	180°

LED's e *buzzer*

Estarão presentes na plataforma vários LEDs e um *buzzer* que permitem comunicar sobre o estado do robô. Três dos LEDs e o *buzzer* podem ser atuados pelo *software* implementado na unidade de controlo, estes podem ser de cores diferentes afim de poder assinalar situações específicas. Por exemplo, um LED amarelo pode ser usado para indicar bateria baixa e um LED vermelho pode ser usado para indicar uma situação de alarme ou erro. Outro LED serve apenas de indicação de que o robô está a ser alimentado, assinalado por *PWR* na Figura 3.22.

Os dois LEDs assinalados por *LEDs ZigBee* são atuados pela solução de comunicação sem fios baseada em ZigBee. O *buzzer* é um dispositivo piezo-elétrico que emite som quando é atuado por um sinal apropriado, deve ser uma onda com a frequência que se quer que seja emitida. A sua posição na plataforma é assinalada na Figura 3.22 por *Buzzer*.

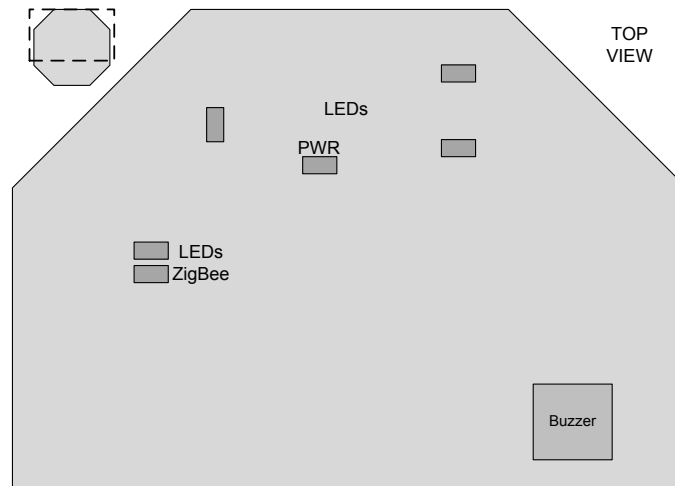


Figura 3.22: Posição dos LEDs e *Buzzer* na plataforma Micro-Rato

3.3.3 Comunicação

Numa primeira fase a comunicação com o robô foi efetuada de forma cablada através de um cabo *Universal Serial Bus* (USB). No robô foi incluído um conversor USB-UART que permite comunicar com uma *Universal Asynchronous Receiver Transmitter* (UART) da unidade de processamento. Este tipo de solução de comunicação foi em primeira instância usada para programar a unidade de controlo, sendo que posteriormente foi usada para controlo do robô.

Para a conversão USB-UART foi usado o conversor FT232R da FTDI [51]. Este conversor permite emular uma porta COM o que elimina a necessidade de desenvolvimento de software específico para lidar com o protocolo USB.

Para potenciar a comunicação foi necessário definir um protocolo de comunicação. A definição deste protocolo pode ser encontrada no Anexo 6.

Era, também, um requisito para esta dissertação que o robô apresentasse uma forma de comunicação sem fios. Tal comunicação devia possibilitar o controlo remoto da plataforma e a receção dos seus dados sensoriais. O levantamento de possíveis tecnologias de comunicação a usar baseou-se nos seguintes requisitos:

- Baixo custo;
- Baixo consumo de energia;
- Facilidade de implementação;
- Comunicação baseada num *standard*;

Dos requisitos de construção da plataforma, pode-se afirmar que não se quer comprometer o preço e o consumo de energia na comunicação, daí os dois primeiros requisitos apresentados.

Os dois requisitos seguintes são determinantes para que, primeiro, o tempo de implementação da solução de comunicação não seja demasiado elevado e, segundo, para se poder ter as vantagens de interoperabilidade e flexibilidade oferecidas por uma solução de comunicação baseada num *standard*.

Na tabela 3.6 vê-se um resumo das principais características de duas possíveis soluções, sendo elas *BlueTooth* e *ZigBee*. Deve-se ter em conta que muitas destas características são qualitativas e dizem respeito à comparação em causa, a principal ideia é tentar perceber qual destas duas se adapta melhor à aplicação em questão. A escolha destas duas soluções para comparação teve a ver com a sua popularidade para as aplicações em questão, é, assim, mais fácil encontrar exemplos da sua aplicação e é mais fácil encontrar no mercado módulos de comunicação facilmente integráveis com a plataforma.

Tabela 3.6: Comparação entre BlueTooth e ZigBee. Fontes: [11] e [12]

Característica	<i>BlueTooth 2.0</i>	<i>ZigBee</i>
Alcance	10-100 m	10-100 m
Frequências de operação	2,4 GHz	868/915 MHz, consoante a região e 2,4 GHz
<i>Throughput</i> de dados	3 Mbps	20/40 Kbps a 868/915 MHz respectivamente, 250 Kbps a 2,4 GHz
Consumo de energia	Maior	Menor
Número de dispositivos na mesma rede	8	2^{16}

Com base na Tabela 3.6 podem-se tirar várias conclusões imediatas. A solução de comunicação baseada em *Zigbee* tem um consumo menor de energia e permite um maior número de dispositivos na mesma rede o que traz vantagens quando se pretende controlar mais do que 8 dispositivos. Já a maior taxa de transmissão de dados do *Bluetooth* traz vantagens quando se pretender transmitir maiores quantidades de dados.

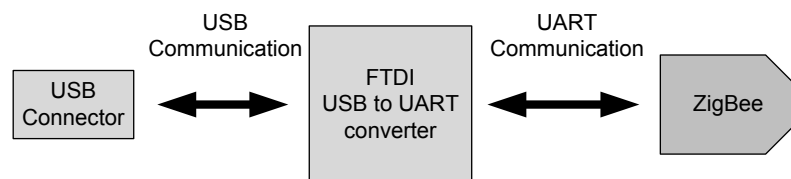


Figura 3.23: Esquema do módulo de interface *ZigBee* com a entidade externa

Para plataforma robótica desenvolvida escolheu-se comunicação baseada em *ZigBee* uma vez que esta abre maior possibilidade em termos de futura expansão da rede e apresenta consumos mais reduzidos. O módulo de comunicação usado foi o *XBee*, XB24-ACI-001 da DIGI [52]. Este módulo implementa a *stack ZigBee* num microcontrolador e permite-nos interagir com ele através de comandos AT. A comunicação com a unidade de controlo da plataforma é efetuada através de uma UART sendo que os níveis lógicos desta são definidos pela tensão de alimentação do módulo, que pode variar entre 2,8 e 3,4 V. Por omissão o

baudrate da comunicação é 9600 baud. De acordo com o manual destes módulos deve-se conseguir comunicação com um alcance de 30 m *indoor* e 90 m *outdoor*.

Tabela 3.7: Sequências de comandos para configuração de um link permanente entre dois módulos *XBee*

Módulo 1	Módulo 2	Descrição
+++	+++	Esta sequência permite colocar o módulo <i>XBee</i> em modo de comando, o que for enviado subsequentemente será interpretado como comandos de configuração do módulo. A resposta deve ser a sequência de caracteres OK seguida de <CR>.
ATRE<CR>	ATRE<CR>	Esta sequência permite restaurar as definições de fábrica dos módulos <i>XBee</i> , desta forma, garante-se que as definições de ambos os módulos são iguais e eles vão comunicar após efetuar os comandos seguintes. A resposta deve ser OK seguido de <CR>
ATMY0<CR>	ATMY1<CR>	Define o endereço do módulo, neste caso quer-se que ambos tenham endereços diferentes. A resposta deve ser OK seguido de <CR>
ATDL1<CR>	ATDL0<CR>	Definem-se os endereços de destino dos módulos como o endereço do outro módulo. A resposta deve ser OK seguido de <CR>
ATWR<CR>	ATWR<CR>	As definições anteriormente programadas são agora escritas em memória não volátil. Após este comando pode-se desligar os módulos <i>XBee</i> da fonte de alimentação que eles irão comunicar entre eles da próxima vez que forem ligados. Novamente deve-se obter como resposta OK seguido de <CR>.

Dada a necessidade de haver uma entidade externa a comunicar com o robô tem de ser desenvolvida uma solução que permita implementar uma comunicação entre estas duas entidades. Para isso parte-se do pressuposto que a entidade externa possui comunicação USB e foi desenhada a solução que pode ser esquematizada na Figura 3.23 e cuja implementação elétrica pode ser encontrada no Anexo 2. Para fazer a conversão entre USB e a UART necessária para comunicar com o módulo *XBee*, utilizou-se o conversor FTDI já referido anteriormente. Usando esta solução é possível interagir com os módulos *XBee* que implementam a comunicação como se se estivesse a lidar com uma porta série simplificando o desenvolvimento da solução de comunicação.

A configuração de uma ligação resume-se a uma pequena série de comandos que pode ser feita apenas uma vez e depois memorizada em memória não volátil do microcontrolador, o que significa que a ligação será restabelecida entre os dois módulos mesmo depois de estes serem desligados da fonte de alimentação.

Para configurar estes módulos usa-se uma sequência definida de caracteres que devem ser enviados através da UART para que estes saibam que a partir daí devem interpretar os dados

recebidos como comandos. Esta sequência é formada por três caracteres '+' seguidos, sendo que o módulo deve responder com a sequência de caracteres *OK* seguido de *carriage return*. A execução bem sucedida de um comando deve ter sempre uma resposta por parte do módulo *XBee*, podendo esta ser *OK* seguido de *carriage return* ou, no caso do comando definir um pedido de informação ao módulo, essa mesma informação seguida de *carriage return*. Para saber mais informações sobre que comandos se podem enviar deve-se consultar o documento referido em [52].

Na Tabela 3.7 vê-se a sequência de comandos que pode ser usada para estabelecer um link permanente entre dois módulos *XBee*. Depois desta configuração ser feita uma vez não há necessidade de a voltar a repetir. Na tabela referida, <CR> significa o carácter *carriage return*, 0x0D em hexadecimal.

Para além das possibilidades já referidas, para cumprir com a necessidade de a plataforma ser expansível deve existir uma forma de as possíveis expansões poderem comunicar com a unidade de controlo. Para isto estão posicionadas na plataforma várias interfaces de comunicação contemplado diferentes tecnologias de comunicação. Em específico *Inter-Integrated Circuit* (I2C), SPI e *Controller Area Network* (CAN) em posições que podem ser vistas na Figura 3.24.

Na plataforma robótica, as posições dos diversos módulos que implementam a comunicação podem ser vistas na Figura 3.24.

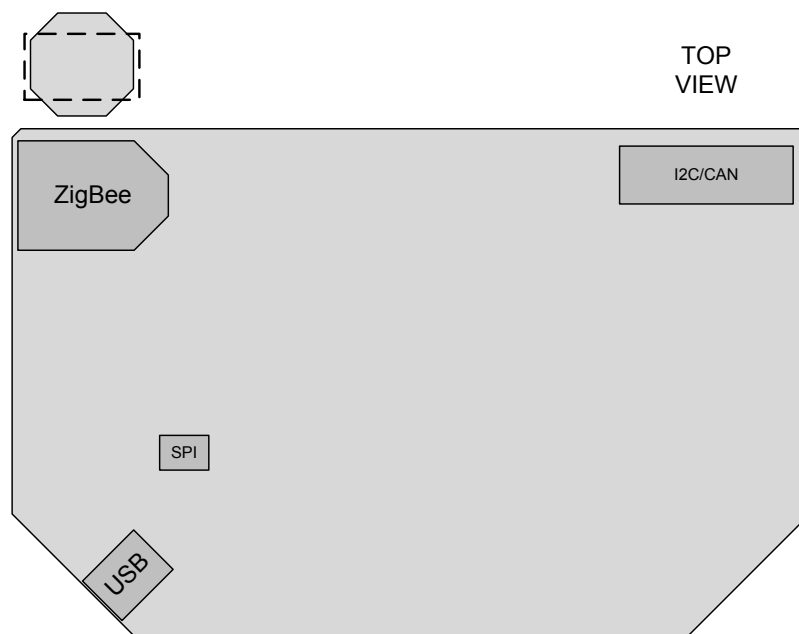


Figura 3.24: Posição das diferentes partes da comunicação na plataforma. Vista de topo

3.3.4 Suporte para processamento

Uma unidade muito importante da plataforma robótica é a sua unidade de processamento. Do que foi sendo dito anteriormente no que diz respeito ao suporte sensorial, à atuação e à comunicação podemos inferir vários requisitos de funcionalidades que a unidade de controlo deve possuir:

- Para a deteção de obstáculos a unidade de controlo deve possuir uma ADC com pelo menos 3 entradas, o que permitirá converter o sinal analógico gerado pelos sensores num sinal digital. Adicionalmente para se poder desligar estes sensores, poupando assim energia, necessitaremos de uma linha digital.
- Dos sensores que permitem distinguir entre zonas refletoras e não refletoras infere-se a necessidade da unidade de controlo possuir mais 7 linhas digitais, 5 delas para poder obter os valores das leituras dos sensores e duas para poder ligá-los ou desligá-los (há duas séries, uma de 3 e outra de 2 sensores).
- Para a deteção do farol de infravermelhos necessita-se de um linha digital para ler o valor de saída e outra linha digital para atuar sobre o servo-motor com capacidade para gerar um sinal PWM.
- Dos *encoders* infere-se a necessidade de mais 4 linhas digitais (2 *chip select*, *clock* e saída digital de dados).
- Necessita-se de mais 3 linhas digitais para ler os valores dos botões presentes no robô.
- Mais à frente será abordado um sensor para monitorizar a tensão da bateria que exigirá mais três entradas de ADC.
- Para a locomoção do robô, a interface com a ponte-H necessita de mais 7 linhas digitais: para cada motor são necessários dois sinais para controlar a direção de rotação e um sinal de PWM e pode-se também aproveitar a existência de uma entrada de *Stand By* na ponte-H para a desligar quando esta não necessitar de ser usada.
- Para os LEDs e para o *Buzzer* necessita-se de mais 4 linhas digitais.
- Para a comunicação necessita-se de duas UART, uma para o Conversor USB UART da FTDI e outra para comunicar com o módulo de comunicação sem fios. Também deve estar disponível comunicação SPI, I2C e CAN.

Resumem-se os requisitos mínimos necessários na Tabela 3.8.

Tabela 3.8: Características mínimas da unidade de controlo

Recurso	Número
Linhas Digitais (excluindo PWM)	27
Entradas Analógicas	6
Saídas PWM	3
Comunicação	2 x UART, SPI, I2C e CAN

Nos vários projetos desenvolvidos na Universidade de Aveiro existe uma grande tradição de utilização de microcontroladores PIC da *Microchip*. Já se teve a oportunidade de ver exemplos da sua utilização no Capítulo 2 quando se referiram as plataformas robóticas didáticas desenvolvidas na Universidade de Aveiro. Esta empresa disponibiliza um Microcontrolador baseado na arquitetura MIPS de 32bits chamada PIC32 com as seguintes características principais [53]:

- Até 512 kbytes de memória Flash;
- Até 128 kbytes de memória *Random-access Memory* (RAM);
- Velocidades de *clock* até 80 MHz;

Existindo uma grande experiência no grupo de trabalho com microcontroladores da *Microchip* conhecem-se as soluções que se apresentam em termos de ferramentas de desenvolvimento o que faz com que haja uma grande familiaridade com estes microcontroladores. A utilização de um microcontrolador PIC na plataforma robótica vem permitir também a continuidade de trabalhos anteriores.

Estes microcontroladores apresentam um conjunto alargado de periféricos e em termos de comunicação com o exterior pode-se encontrar desde Ethernet, CAN, USB a SPI e I2C.

Dentro da família de microcontroladores PIC32 escolheu-se o PIC32MX795F512H cujas características principais são resumidas na Tabela 3.9 [54].

Tabela 3.9: Características do microcontrolador PIC32MX795F512H

Característica	Descrição
Número de linhas de I/O digital	53
Número de canais analógicos (10 bits/1 Msps)	16
Saídas PWM	5
Comunicação	6 x UART, 4 x SPI, 5 x I2C, 2 x CAN, USB e Ethernet

Por comparação das Tabelas 3.8 e 3.9 conclui-se que este microcontrolador tem a capacidade de suprir as necessidades apresentadas. Não se deve esquecer, como é avisado em [54], que alguns dos 64 pinos do *package* que se escolheu têm várias funções multiplexadas, o que significa que pode haver dificuldades em conseguir que usar tudo o que se pretende ao mesmo tempo. Fazendo a alocação dos pinos do microcontrolador à função que cada um deveria desempenhar surgiu o problema de estes não serem suficientes para que se pudesse usar em simultâneo, SPI e a UART de interface com o módulo de comunicação sem fios. Assim, deve-se ter em conta que só se pode usar comunicação SPI quando não se estiver a usar comunicação sem fios.

Para completar os dados da Tabela 3.9 deve-se ainda dizer que este microcontrolador possui 512 kB de memória Flash, 128 kB de memória RAM e velocidades de CPU que podem ir até um máximo de 80 MHz. Desta forma abre-se bastante o leque de possibilidades oferecidas por este microcontrolador, não se limitando pela unidade de controlo as possibilidades de quem for desenvolver aplicações para a plataforma robótica.

3.3.5 Alimentação da plataforma

Por ser uma forma prática e de fácil implementação a plataforma será alimentada a partir de baterias. Para não haver grandes limitações ao tipo de bateria a usar decidiu-se preparar a plataforma quer do ponto de vista mecânico quer do ponto de vista elétrico para poder ser compatível com vários tipos de baterias.

Para haver bastante liberdade na escolha da bateria em termos elétricos implementou-se um esquema de regulação de tensões que pode ser mais bem entendido com a ajuda da Figura 3.25. Aqui vê-se que a bateria pode ser escolhida de forma a apresentar uma tensão entre 3 e 15 V. Para baterias com tensões inferiores a 10 V é usado um regulador *step-up* que eleva a tensão para 10 V, já uma tensão superior a 10 V alimenta diretamente os motores de locomoção sendo o regulador *step-up* dispensado. A tensão mínima é limitada pelo regulador *step-up* e a tensão máxima é limitada pela menor das tensões máximas dos vários componentes ligados diretamente à bateria. Esta limitação é imposta pela ponte-H que suporta uma tensão máxima de 15 V [48]. Deve ter-se em atenção que para escolher entre alimentar o robô com menos ou mais de 10 V se deve alterar fisicamente as ligações à eletrônica da plataforma. Para isso existem 4 *jumpers* devidamente assinalados. Estes *jumpers*, que se podem observar na Figura 3.25, chamados J_1 , J_2 , J_3 e J_4 , permitem configurar a plataforma para a tensão da bateria. Fechando J_3 , J_4 e J_1 e deixando J_2 aberto pode-se alimentar a plataforma entre 3 e 10 V. O Inverso (fechar J_2 deixando os outros três abertos) permite alimentar a plataforma entre 10 e 15 V.

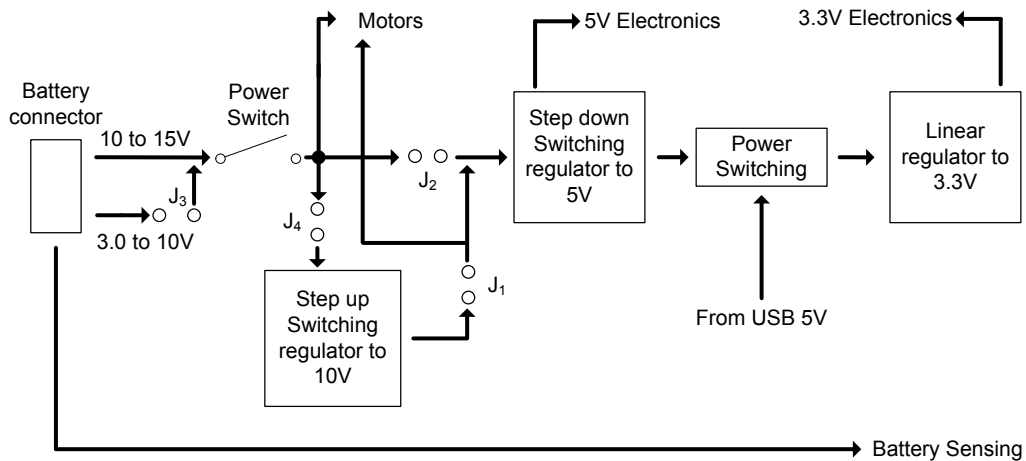


Figura 3.25: Esquema da unidade de alimentação da plataforma

Para conhecer o estado das baterias implementou-se um esquema de medição das tensões das mesmas. Este esquema baseou-se num divisor resistivo para poder medir a tensão no intervalo de leitura das ADCs da unidade de controlo seguido de um amplificador operacional para baixar a resistência de saída do sensor. Contemplámos um esquema que pudesse medir até 3 tensões diferentes. Em alguns tipos de baterias, como são exemplo as baterias de polímeros de lítio e iões de lítio com várias células, é necessário monitorizar cada célula individualmente para garantir que a tensão de nenhuma delas desce abaixo de um certo valor, o que, a acontecer, poderia causar danos irreversíveis à bateria [55]. O esquema de medição de tensões foi implementado como pode ser visto da Figura 3.26. Neste esquema vêem-se assinalados como 1S, 2S e 3S as saídas da primeira, da segunda e da terceira células, respetivamente e como OUT1, OUT2 e OUT3 as saídas dos respetivos sensores para a unidade de controlo. Tendo como referência uma bateria de lítio de três células, as tensões destas variam, entre o valor mínimo e máximo, de 3 a 4,2 V, de 6 a 8,4 V e de 9 a 12,6 V para a primeira, a segunda e a terceira célula, respetivamente. Com o esquema implementado, obtém-se nas saídas OUT1, OUT2 e OUT3, uma variação entre 0,77 e 1,08 V, 0,75 e 1,05 V

e 0,77 e 1,07 V, respetivamente. Para a obtenção destes valores usou-se a Expressão 3.3.

$$V_{out} = V_{in} \times \frac{R_l}{R_l + R_u} \quad (3.3)$$

Onde V_{out} é a tensão nas saídas OUT1, OUT2 e OUT3, V_{in} é a tensão das células e as resistências R_l e R_u são as resistências que se encontram entre a entrada do seguidor de tensão e a massa e entre a entrada do sensor e a entrada do seguidor de tensão, respetivamente.

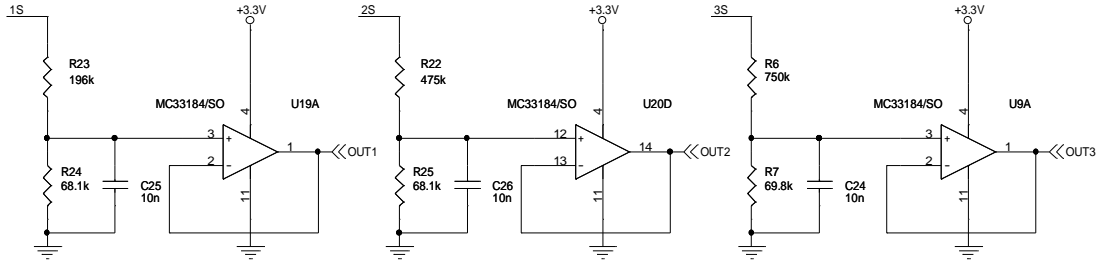


Figura 3.26: Esquema do circuito do monitorização do nível das baterias

A eletrónica de mais baixa tensão (3,3 V), pode ser alimentada a partir da ligação USB o que permite operações básicas, principalmente de interação com a unidade de controlo. Pode assim fazer-se algum *Debug* de *Software* sem necessidade de ter uma bateria acoplada ao robô. Para fazer a comutação e a escolha entre qual das fontes de energia vai alimentar o robô quando as duas estão ligadas simultaneamente, implementou-se um esquema baseado no circuito da Figura 3.27 e decidiu-se que na presença da bateria esta alimentaria toda a plataforma. O Funcionamento do esquema apresentado baseia-se no pressuposto de que o diodo D_2 vai apresentar uma tensão V_D que será inferior à do diodo D_1 por o primeiro ser um diodo de *Schottky*, que apresenta uma tensão V_D de, aproximadamente, 0,3 V, por oposição a aproximadamente 0,7 V para diodos de silício [56]. Assim, como as tensões V_{USB} e V_{REG} estão ambas muito próximas de 5 V, no cenário estarem ambas ligadas a tensão à entrada do regulador de 3,3 V vai ser de $5 - V_{D2}$ V o que não é suficiente para que o diodo D_1 esteja em condução fazendo com que a plataforma seja alimentada a partir da bateria.

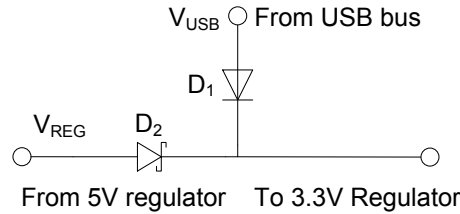


Figura 3.27: Circuito de comutação da alimentação das tensões da eletrónica de baixa tensão

Antes de partir para a escolha dos reguladores usados convém saber quais as correntes envolvidas para se projetar os reguladores para as correntes necessárias. Assim, resume-se na Tabela 3.10 os consumos máximos de corrente dos vários componentes da plataforma robótica. Os valores foram obtidos dos *datasheets* dos componentes em questão, considerando

as características de operação mais próximas das que se verificam na plataforma. Para o caso do servo-motor não foi possível encontrar na sua documentação referência para o consumo de corrente, por isso usou-se uma estimativa baseada num outro servo-motor que se pode encontrar em [57], este é um servo-motor capaz de gerar um torque maior do que o daquele que se utilizou, pelo que, se pode admitir que é uma estimativa que não peca por defeito. No caso do módulo *XBee* o consumo deste varia conforme esteja a receber, a enviar dados ou e, repouso, por isso considerou-se o caso em que está a receber e a enviar ao mesmo tempo.

Tabela 3.10: Consumo de corrente das várias partes da plataforma robótica

Diretamente da tensão da bateria		5V		3,3 V	
Componente	$I(mA)$	Componente	$I(mA)$	Componente	$I(mA)$
Motores(cada um)	687	Encoders	20	Ponte H	1,8
		Servo Motor	180	Sensor para deteção do farol de infra-vermelhos	0,6
		Buzzer	5	XBee	75
		Emissor de Infravermelhos	60	PIC32	70
		Sensores de distância (cada um)	40	<i>Transceiver</i> CAN	70

Dos dados da tabela anterior pode-se então calcular a necessidade de corrente para cada um dos reguladores. Começando pelo regulador linear de 3,3 V, é imediato inferir que necessitamos de um regulador com uma corrente máxima superior a 147,4 mA que é a soma das correntes de todos os dispositivos que estão ligados a esta tensão. Já para 5 V necessita-se da corrente que vai para o regulador de 3,3 V mais 385 mA que é a soma das correntes consumidas pelos componentes ligados a 5 V.

Para os reguladores de 3,3 V e 5 V teve-se o cuidado de usar um sobredimensionamento que fosse suficiente para acomodar futuras expansões da plataforma. Isto para que essas expansões pudessem retirar a alimentação diretamente da plataforma em vez de usarem a sua própria alimentação.

Para o regulador para 3,3 V escolheu-se o TC1262 da Microchip [58]. É um regulador linear de *Low Dropout*. Pode ser alimentado até uma tensão mínima de 650 mV acima da tensão de saída, ou seja, 3,95 V. Este regulador tem capacidade para uma corrente máxima de 500 mA o que está acima das necessidades. Assim, está disponível uma corrente de aproximadamente 350 mA para as futuras expansões alimentadas a 3,3 V.

Uma vez que se sobredimensionou o regulador linear para 3,3 V deve-se projetar o regulador para 5 V de forma a contemplar esse sobredimensionamento. Assim este deve ser projetado para um mínimo de corrente de 885 mA. Este é um regulador de *switching* o que significa que o seu rendimento vai ser maior que o do regulador usado para 3,3 V. Deve-se lembrar que para o caso do regulador para 3,3 V a diferença de tensão no conjunto díodo mais regulador vai ser de 1,7 V, ao passo que para o regulador para 5 V essa diferença pode chegar até 10 V para

o caso de alimentarmos a plataforma com a tensão máxima que é de 15 V. Assim justifica-se que este regulador seja de *switching* pois, no caso de ser linear, iria dissipar muito mais energia que o regulador para 3,3 V. Escolheu-se o regulador OKI-78SR da Murata [59] que é uma solução que implementa um regulador *step-down* completo. Este regulador é capaz de fornecer até 1,5 A sendo que, com essa corrente e com uma tensão de 12 V na entrada, apresenta um rendimento de aproximadamente 85 %. Usando este regulador sobram 615 mA para futuras expansões.

Regulador *Step-up*

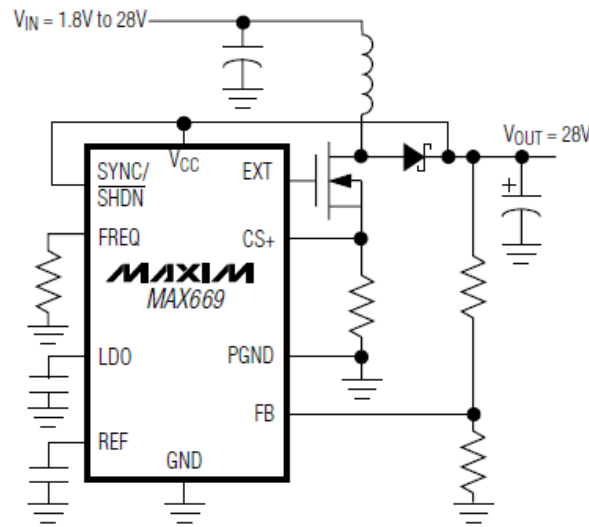


Figura 3.28: Circuito tipo para implementar um regulador *step-up* baseado no chip MAX669

Entra-se agora no projeto do regulador *step-up* para 10 V. Em primeiro lugar deve notar-se que o sobredimensionamento deste regulador iria trazer desvantagens pois exigiria um dimensionamento exagerado de alguns componentes aumentando o custo e o tamanho ocupado pela solução. Deve, então, fazer-se um dimensionamento de acordo com as necessidades reais. Sendo assim é necessário que o regulador para 5 V discutido no parágrafo anterior forneça 532,4 mA, que é a soma das correntes que são necessárias para alimentar a eletrónica ligada aos 5 V com a corrente necessária para o regulador linear para 3,3 V. Com esta corrente pode supor-se que o rendimento do regulador para 5 V é próximo de 75 %, o que é um valor conservativo para garantir que não há um subdimensionamento do regulador *step-up*. Pode aplicar-se a Expressão 3.4 chegando ao valor de 354,9 mA, ou seja, no total, deve projetar-se o circuito para 1,73 A, valor obtido somando a corrente máxima consumida pelos motores.

$$I_{in} = \frac{V_{out} \times I_{out}}{\eta_{regulador} \times V_{in}} \quad (3.4)$$

Escolheu-se o *chip* MAX669 da Maxim para o projeto do regulador para 10 V [60] e usou-se a ferramenta de desenho e simulação EE-sim [61] para ajudar no projeto. Este *chip* regula a tensão com a ajuda de um transístor externo o que permite trabalhar com correntes mais elevadas. O circuito tipo de um regulador *step-up* usando este regulador pode ser visto na

Figura 3.28. Nesta configuração o *chip* é alimentado com a tensão de saída o que leva a que a tensão de entrada possa variar desde o mínimo imposto pelo *chip*, que é 1,8 V até ao valor da tensão de saída [60].

Para o projeto do circuito inseriu-se no EE-sim os requisitos necessários, que foram os que mostra a Tabela 3.11. Deve ter-se em conta que o *ripple* da tensão de saída não é muito crítico uma vez que este regulador apenas alimenta diretamente os motores.

Tabela 3.11: Consumo de corrente das várias partes da plataforma robótica

Requisito	Valor
Tensão de entrada(intervalo)	3 – 10 V
Tensão de saída	10 V
Ripple da tensão de saída	0,1 V
Corrente de saída	1,73 A

A partir do resultado da simulação surgiu o circuito da Figura 3.29

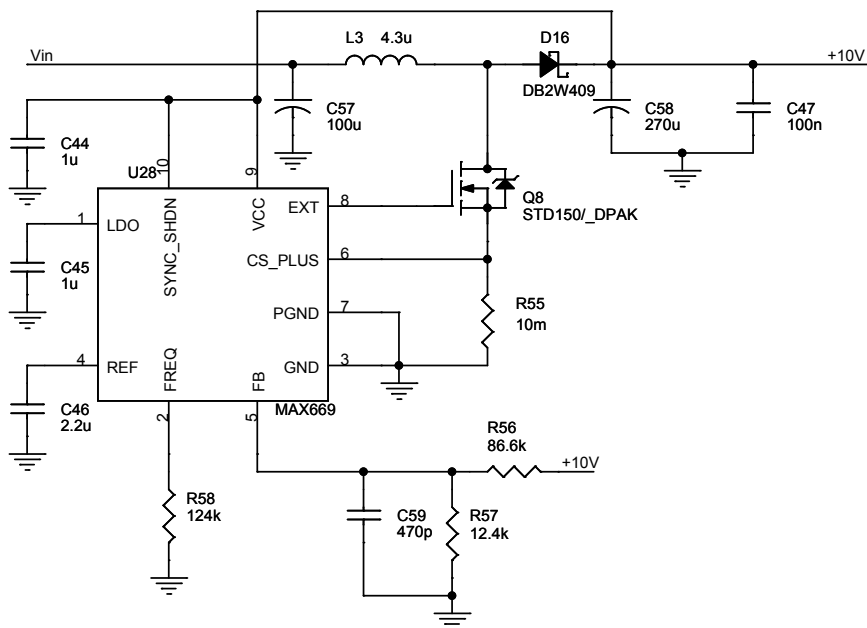


Figura 3.29: Circuito para implementar um regulador *step-up* baseado no chip MAX669

A escolha dos componentes deve ser cuidada, por isso remete-se a [60] como documento de referência para se conhecer as características que se procuram para cada um dos componentes. Apresenta-se, de seguida, a lista das características requeridas para os componentes chave do circuito:

- Para os componentes *C44*, *C45*, *C46*, *R58*, *C59*, *R57*, *R56* e *C47* as suas características podem ser facilmente encontradas no *datasheet* do MAX669 [60].

- Os condensadores $C57$ e $C58$, condensadores de entrada e saída respetivamente, devem possuir um valor de Resistência Equivalente em Série o mais baixa possível. No caso do condensador $C58$ isto ajuda a diminuir o *ripple* da tensão de saída e a aumentar a eficiência pois diminui as perdas, este último efeito acontece também no condensador de entrada $C57$. Para o condensador de entrada usou-se um condensador de alumínio de $100\mu\text{F}$ com $10\text{m}\Omega$ de Resistência Equivalente em Série e para o condensador de saída um condensador de alumínio com $470\mu\text{F}$ de capacidade e $20\text{m}\Omega$ de Resistência Equivalente em Série.
- A bobina $L3$ deve possuir baixa resistência DC para que as perdas sejam menores e deve possuir uma corrente de saturação suficiente para conseguir conduzir o máximo de corrente necessário. O valor da indutância desta bobina é dimensionado para o máximo de corrente de forma a garantir a estabilidade do circuito. Uma vez que foi difícil encontrar uma bobina com o valor de indutância definido pelo simulador, que era $3,6\mu\text{H}$, e com as outras características necessárias, escolheu-se o valor mais próximo encontrado, que foi $4,3\mu\text{H}$ numa bobina com uma resistência DC de $17\text{m}\Omega$ e uma corrente de saturação de 8A . Na Figura 3.30 pode ver-se uma simulação efetuada com o EE-Sim em que se inseriu estes valores para a indutância da bobina e 3V para V_{in} por este valor levar ao máximo de corrente conduzido pela bobina. Pode notar-se que o máximo de corrente, que é $6,82\text{A}$, é inferior à corrente de saturação da bobina. As características simuladas são as mesmas do circuito físico apresentado na Figura 3.29.

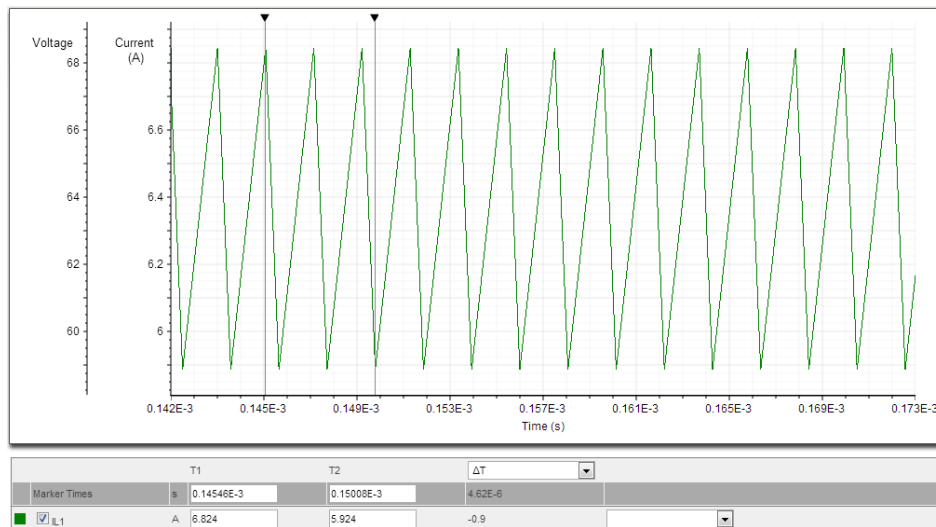


Figura 3.30: Forma de onda da corrente na bobina $L3$, frequência de *switching* de 500kHz

- O diodo $D16$ deve ser capaz de conduzir uma corrente igual ou superior à máxima corrente de saída do circuito. Deve, também, ter também um tempo de recuperação rápido. Escolheu-se um diodo que tem capacidade para conduzir até 3A de corrente média e 25ns de tempo de recuperação.
- O transistor $Q8$ deve ser escolhido de forma a que a sua resistência $R_{ds(ON)}$ seja a menor possível e que a sua capacidade de entrada C_{gs} seja também a pequena. A resistência vai fazer com que haja perdas por efeito de Joule quando o transistor está

em condução. A capacidade de entrada vai ser responsável pela condução de corrente do e para o MAX669 quando há comutação de estado e vai também aumentar os tempos de comutação. Também é necessário que este transistor seja capaz de conduzir uma corrente maior que a corrente de pico da bobina. Nesse sentido escolheu-se um transistor com $R_{ds(ON)}$ de $2,8\text{ m}\Omega$, C_{gs} de $3,7\text{ nF}$ e capacidade para conduzir até 80 A de corrente de pico.

- Finalmente, a resistência $R1$ vai permitir que o circuito se desligue caso esteja a consumir mais corrente que o permitido. Escolheu-se uma resistência de $10\text{ m}\Omega$ que pode dissipar até 1 W de potência de pico.

Para finalizar, efetuou-se uma simulação do circuito do regulador *step-up* usando a ferramenta EE-Sim. Simulou-se para o caso em que este está a consumir a corrente máxima e supondo na entrada a tensão mínima de alimentação que é de 3 V , esta produz o mínimo de rendimento, podemos chegar a uma estimativa da potência máxima consumida pela plataforma e do rendimento do próprio regulador *step-up*. Estes são, respetivamente, $18,6\text{ W}$ e 93% .

3.4 Expansibilidade

Para o futuro uso da plataforma esta foi pensada de forma a poder ser expandida com novas funcionalidades. Contemplaram-se duas formas de expansão. A primeira pressupõe que a expansão vai ser realizada diretamente na plataforma, ou seja, o que for adicionado vai ser adicionado à PCB que constitui a base do robô como se se tratasse de um componente desta. A segunda forma de expansão pressupõe que se vai desenvolver uma solução à parte que depois vai ser acoplada à plataforma, ficando numa base elevada em relação à base do robô.

Para o primeiro caso foram previstas na plataforma 3 áreas de prototipagem que se caracterizam por serem um array de furos espaçados entre eles de $2,54\text{ mm}$, permitindo adicionar diretamente à plataforma componentes cuja distância entre pinos seja essa. Estas 3 áreas possuem também alimentação com as várias tensões presentes na plataforma para que possam alimentar as soluções que o utilizador quiser adicionar.

Deve ter-se presente que ao alimentarmos as expansões adicionadas a partir da plataforma apenas o podemos fazer quando a tensão da bateria for superior a 10 V não usando, desta forma, o circuito *step-up*. Devemos garantir-se que o consumo do que adicionarmos não vai exceder 350 mA a $3,3\text{ V}$ e 615 mA a 5 V .

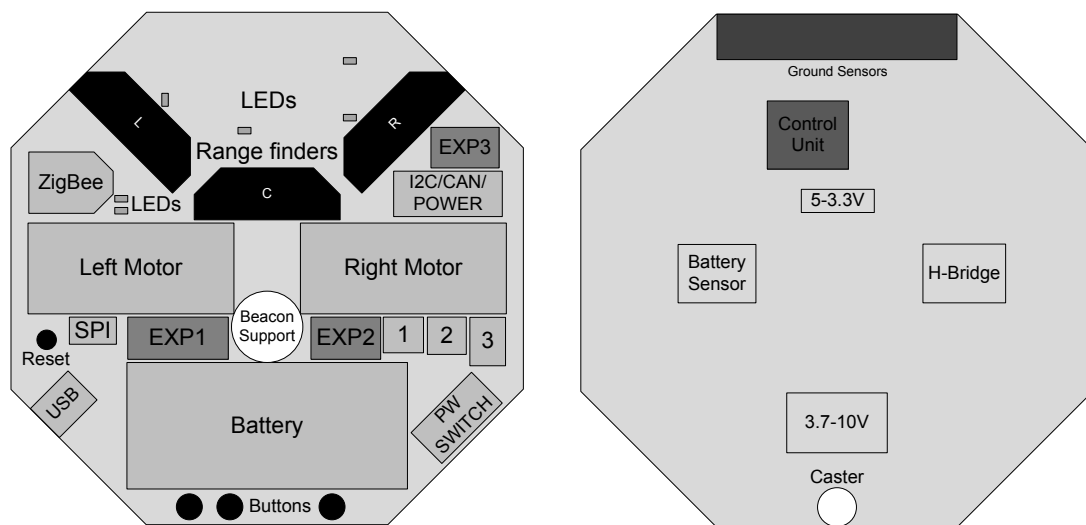
No caso do segundo tipo de expansão para que as soluções desenvolvidas possam ser adicionadas à plataforma foram incluídos na PCB desta vários furos para que o acoplamento possa ser realizado. Estes furos foram colocados de forma a cobrirem a maior área possível.

Para se obter mais informação sobre as áreas de expansão e a posição dos furos na plataforma pode consultar-se o Anexo 4.

É importante realçar que estas expansões podem ser ligadas à unidade de controlo através das várias interfaces de comunicação que foram previstas na plataforma e que podem ser vistas na Figura 3.24.

3.5 Realização Física

Tem-se vindo a focar cada funcionalidade da plataforma individualmente. Agora, deve passar-se a encará-la como um todo e observar como tudo se encaixa. Nas Secções 3.3 e 3.4 foi-se apresentando os locais escolhidos na plataforma para a localização de alguns dos componentes que implementam as funcionalidades mais importantes da plataforma. Na Figura 3.31 pode observar-se as posições dos componentes que implementam cada uma das funcionalidades, desta forma, tem-se uma ideia geral de onde vão ficar colocados. Os componentes assinalados com 1,2 e 3 representam, respetivamente, o regulador de *switching* para 5 V o *buzzer* e uma ficha de programação que permite programar, na unidade de controlo, um *bootloader* que faz com que a partir daí se possa utilizar a interface USB para programação, através do conversor USB- UART.



(a) Posição dos componentes na plataforma, vista de topo

(b) Posição dos componentes na plataforma, vista de baixo

Figura 3.31: Posição dos diversos componentes na plataforma robótica

Como já se referiu várias vezes a base da plataforma é uma PCB que serve de estrutura para todo o robô.

No âmbito desta dissertação foi apenas construída uma plataforma que serviu de protótipo para validar o projeto.

3.6 Software de suporte

O software de suporte que é aqui referido contempla tanto aquele que foi implementado no robô como na entidade externa que comunica com este. Este software pode ser dividido em quatro grupos, consoante a sua funcionalidade, sendo estes:

- **API de abstração da camada física** implementada no robô para interagir com este através de um conjunto de funções em linguagem C sem ter de conhecer nem os pormenores da unidade de controlo nem as especificidades dos sensores e atuadores presentes.

- **Implementação do protocolo de comunicação com a entidade externa** na plataforma robótica para que esta possa interagir com a primeira e ser comandada por esta.
- **Implementação do protocolo de comunicação com a entidade externa** na entidade externa para criar uma forma da interagir e comandar o robô a partir desta.
- Implementação de uma **API na entidade externa** com uma interface semelhante aquela que foi implementada na plataforma robótica.

O propósito deste *software* é permitir aos futuros utilizadores da plataforma servirem-se deste para desenvolverem as suas soluções. Pretende construir-se uma interface o mais completa possível com o *hardware* da plataforma de forma a poder interagir com todas as suas funcionalidades. Todo o software foi desenvolvido em linguagem de programação C.

Na Figura 3.32 pode ver-se um esquema do software que se implementou. Este esquema refere-se ao sistema completo, incluindo a entidade externa. Analisando o esquema, do bloco mais à direita para o bloco mais à esquerda, vê-se, no primeiro bloco, a referência à aquisição de dados da plataforma. Os dados recolhidos podem ser usados ao nível da plataforma robótica através da API que aí foi implementada. Esta também pode ser usada para atuar sobre a plataforma. Servindo-se da API da plataforma foi implementado o protocolo de comunicação nesta e foi também implementado na entidade externa, o que se pode ver nos blocos *Data to transmit* e *Data Received*. Na entidade externa existe um conjunto de estruturas de dados que servem para que se mantenha o registo dos dados sensoriais e de atuação. O utilizador pode usar a *API* implementada na entidade externa para desenvolver as suas aplicações que permitam controlar a plataforma e conhecer os seus dados sensoriais.

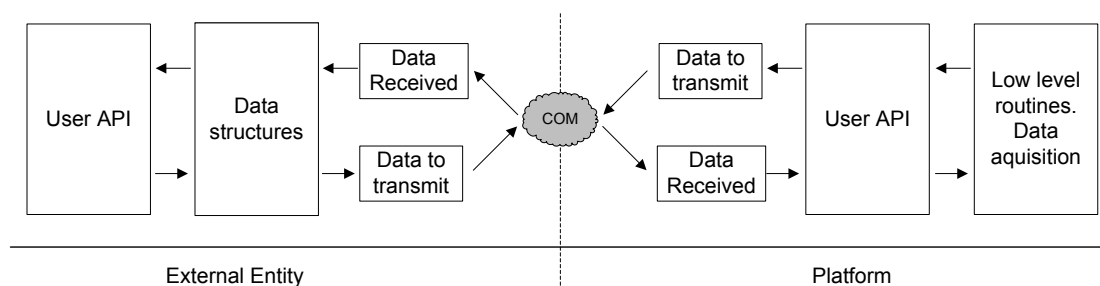


Figura 3.32: Estrutura global do software desenvolvido

API de abstração da camada física da plataforma

Esta API constituída por um conjunto de funções em C permite implementar o software de controlo servindo-se delas para interagir com a camada física da plataforma.

Para os sensores da plataforma optou-se por criar apenas uma função que faz a leitura de todos os dados sensoriais atualizando uma estrutura de dados com os mais recentes valores de todos os sensores. Esta função despoleta todas as conversões necessárias para transformar os dados que são lidos diretamente dos sensores nos dados que vão residir na estrutura de dados.

É também esta função que despoleta todas as leituras analógicas necessárias e se encarrega do processo descrito anteriormente na Secção 3.3.1 para a leitura dos sensores de superfície.

O protótipo desta função é o seguinte:

```
void readSensors()
```

Não são necessários parâmetros de entrada nem a função devolve nada, apenas atualiza a estrutura de dados. Esta está definida como é mostrado a seguir com cada um dos campos resumido em comentário.

```
typedef union
{
    struct
    {
        // Sensor de obstáculos central
        int obst_sens_center;
        // Sensor de obstáculos da direita
        int obst_sens_right;
        // Sensor de obstáculos da esquerda
        int obst_sens_left;
        // Sensor de bateria da 3ª célula
        int bat_sens3;
        // Sensor de bateria de 2ª célula
        int bat_sens2;
        // Sensor de bateria de 1ª célula, os valores apresentados para
        // estes sensores (obstáculos e bateria) são os valores diretamente
        // lidos da ADC. O posterior tratamento destes dados fica ao encargo
        // de quem estiver a desenvolver aplicações.
        int bat_sens1;
        // Tempo decorrido até que a tensão dos
        // vários sensores de superfície desceu
        // abaixo do threshold definido
        // o sensor 1 é o sensor mais à esquerda
        // O tempo real é obtido multiplicando os valores abaixo por 50us
        int elTSens1;
        int elTSens2;
        int elTSens3;
        int elTSens4;
        int elTSens5
        // O sensor de superfície número 3 é considerado o sensor de
        // chão e o valor desta variável assume o valor 1 caso o chão
        // seja não refletor e 0 caso seja refletor.
        BOOL ground_sens;
        // Valores binários dos sensores de linha, nos 5 LSB.
        // Do LSB para o MSB, sensor 5 até 1.
        //
        int line_sens;
```

```

    // Odometria da roda esquerda, em número de pulsos, desde que o robô
    // foi inicializado
    int odo_L;
    // Odometria da roda direita, em número de pulsos, desde que o robô
    // foi inicializado, 960 pulsos equivalem a uma volta completa da
    // roda.
    int odo_R;
};
int array[15];
} MR32_SENS

```

Para a atuação sobre a plataforma robótica foi definido um conjunto de funções, estas, apresentadas de seguida, permitem a atuação sobre os vários atuadores presentes na plataforma.

```

//Inicialização do microcontrolador, inicializa todos os recursos necessários
// para o outro software da plataforma
//
// Não tem parâmetros de entrada nem de saída.
//
// Deve ser a primeira função a ser invocada
void initPIC32(void);

// Atuação sobre a velocidade dos motores de locomoção, a velocidade é
// definida como o número de pontos que o motor deve rodar
// durante o tempo de amostragem do compensador
//
// Entrada:
//   leftSpeed - setpoint para o motor esquerdo
//   rightSpeed - setpoint para o motor direito
//
// O setpoint máximo permitido é de 45
void setVel2(int leftSpeed, int rightSpeed);

// Definição da posição do servo-motor que suporta o sensor de farol
//
// Entradas
//   pos - posição do servo motor, intervalo de -15 a 15. Em que 0
//         é a posição central, -15 a posição mais à esquerda e 15 a
//         posição mais à direita.
void setServoPos(int pos);

// Ligar o led ledNr
//
// Entradas
//   ledNr - número do led a ligar
void setLed(int ledNr);

```

```

// Desliga o led ledNr
//
// Entradas
// ledNr - número do led a desligar
void resetLed(int ledNr);

// Faz com que o buzzer apite durante o tempo definido por val
//
// Entradas
// val - tempo durante o qual o Buzzer vai apitar num múltiplo de 80ms
void beep(int val);

```

Para além destas funções existe também um conjunto de macros que permite interagir com alguns sensores ou atuadores que por a sua interface ser demasiado simples, resumindo-se apenas a uma leitura ou a uma escrita num porto digital não se justificava criar uma função para elas. Este conjunto de macros é apresentado de seguida.

```

// Leitura dos botões presentes na plataforma, por compatibilidade com
// o regulamento do concurso Micro-Rato manteve-se a designação de
// start e stop button, o outro botão foi chamado de extra button.
#define startButton()
#define stopButton()
#define extraButton()

// Ligar ou desligar os sensores de obstáculos para permitir poupar energia
// no caso de não se estar a necessitar deles.
#define enableObstSens()
#define disableObstSens()

// Leitura do sensor de farol.
#define readBeaconSens()

```

Com esta interface está definida a forma de interagirmos com a plataforma através do desenvolvimento de software em linguagem de programação C.

Implementação do protocolo de comunicação na plataforma

O protocolo de comunicação, já referido, foi implementado na plataforma por meio de uma aplicação desenvolvida a partir da API de suporte referida na secção anterior. Esta aplicação implementa o protocolo descrito no Anexo 6 na sua completude.

Pode-se descrever a aplicação desenvolvida por meio do diagrama apresentado na Figura 3.33.

Este esquema ajuda a perceber o software que foi desenvolvido para a plataforma com o fim de implementar o protocolo de comunicação. Quando a plataforma é iniciada usa-se a API anteriormente descrita para inicializar a unidade de controlo. São também inicializados os recursos necessários para a comunicação, nomeadamente as UARTs necessárias para interagir

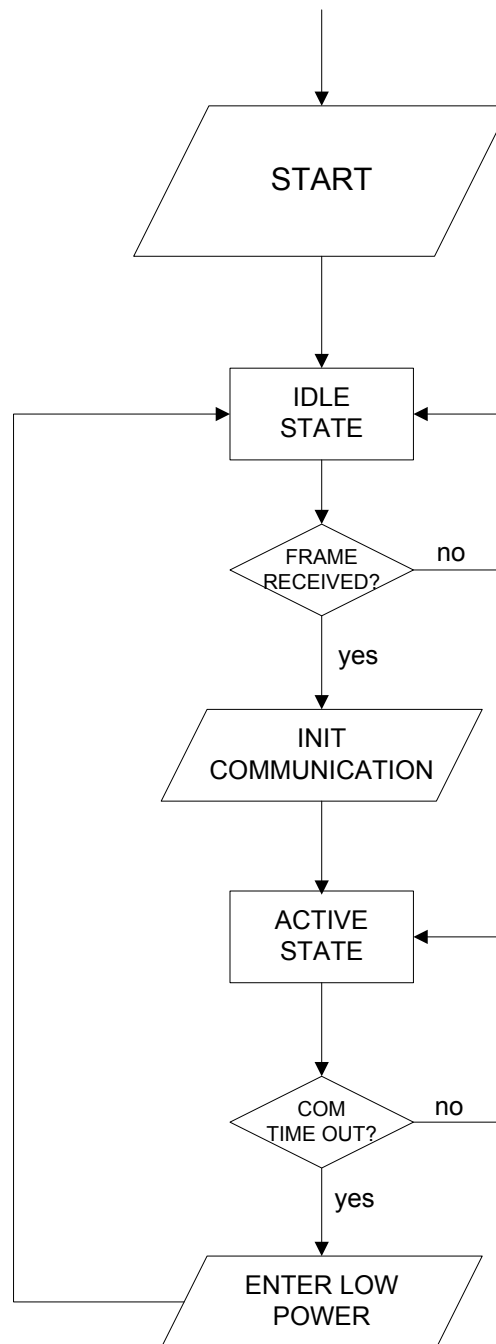


Figura 3.33: Esquema do software que implementa o protocolo de comunicação na plataforma

quer com a comunicação cablada quer com a comunicação *wireless*. Após isso o robô entra no *IDLE STATE* em que se mantêm desligados, ou em modos de poupança de energia, todos os recursos que não estão a ser utilizados e sobre os quais se pode atuar, nomeadamente os sensores de obstáculos e a ponte-H. Após a receção de uma trama válida estes recursos são mudados para modos em que se possa, respetivamente, ler os sensores de obstáculos e atuar sobre os motores de locomoção.

A plataforma mantém a comunicação ativa enviando tramas com informação sensorial a cada 40 ms. Este tempo foi escolhido para ser superior ao período com que os sensores de obstáculos atualizam a sua saída, que é, em média 38,3 ms, como foi dito anteriormente. Estes são os sensores mais lentos da plataforma. Após um período de 0,5 s sem a receção de nenhuma trama a plataforma considera que houve uma falha da comunicação. Após esta ser declarada o robô pára os motores de locomoção, desliga o *buzzer* e os LEDs e é devolvido ao *IDLE STATE* do qual só voltará a sair quando for retomada a comunicação. A declaração de falha de comunicação existe para garantir a segurança da plataforma: se a comunicação falhar a plataforma imobiliza-se.

Uma vez que há duas formas de comunicação, cablada e *wireless*, para a plataforma saber qual será aquela que vai usar para enviar as tramas com informação sensorial escolhe a partir da qual recebeu a última trama. Desta forma pode-se mudar a forma de comunicação sem necessidade de reiniciar a plataforma.

Implementação do protocolo de comunicação na Entidade externa

A entidade externa que comunica com a plataforma robótica materializa-se num PC que corre uma distribuição do sistema operativo Linux. Neste caso em específico o software foi desenvolvido na distribuição Ubuntu.

O software implementado deve servir de base à API de suporte que permitirá comunicar com a plataforma, lendo os seus dados sensoriais e atuando sobre ela. Optou-se por um esquema *multithread* em que se separa a aplicação a ser desenvolvida para interagir com a plataforma do suporte à comunicação. Assim, a interação com a entidade externa é feita de forma independente. O funcionamento da *thread* que implementa a comunicação com a entidade externa pode ser melhor entendido com a observação da Figura 3.34. Esta figura mostra o diagrama de fluxo da *thread* que implementa o protocolo de comunicação, bem como a forma com esta é criada.

Em primeiro lugar, é feita a inicialização de todos os recursos associados à comunicação, o que é indicado por *INIT COM*. Neste caso os recursos de que se falar são as portas série e recursos do sistema operativo que precisem de ser usados para o funcionamento deste *software*. Neste ponto, é também criada a *thread* que a partir daqui controla a comunicação. Quando a *thread* inicia a sua execução os recursos necessários para o seu funcionamento estão já inicializados. Se esta inicialização for bem sucedida a *thread* é então criada e a partir desse momento passa a ser executada em paralelo com o processo principal cujo desenvolvimento vai depender do utilizador. Caso a inicialização falhe vai ser reportado um erro que será comunicado ao utilizador quando este invocar qualquer uma das funções da API. Após a inicialização bem sucedida da *thread*, esta entra no estado assinalado como *ACTIVE COM STATE*, em que inicializa a comunicação enviando uma trama à plataforma para que o *buzzer* presente nesta apite durante 120 ms, assim haverá uma indicação sonora do início bem sucedido da comunicação. A partir deste momento, as ordens para atuação que o utilizador da API der por meio desta serão transmitidas à plataforma. Periodicamente, a cada 40 ms, é enviada uma

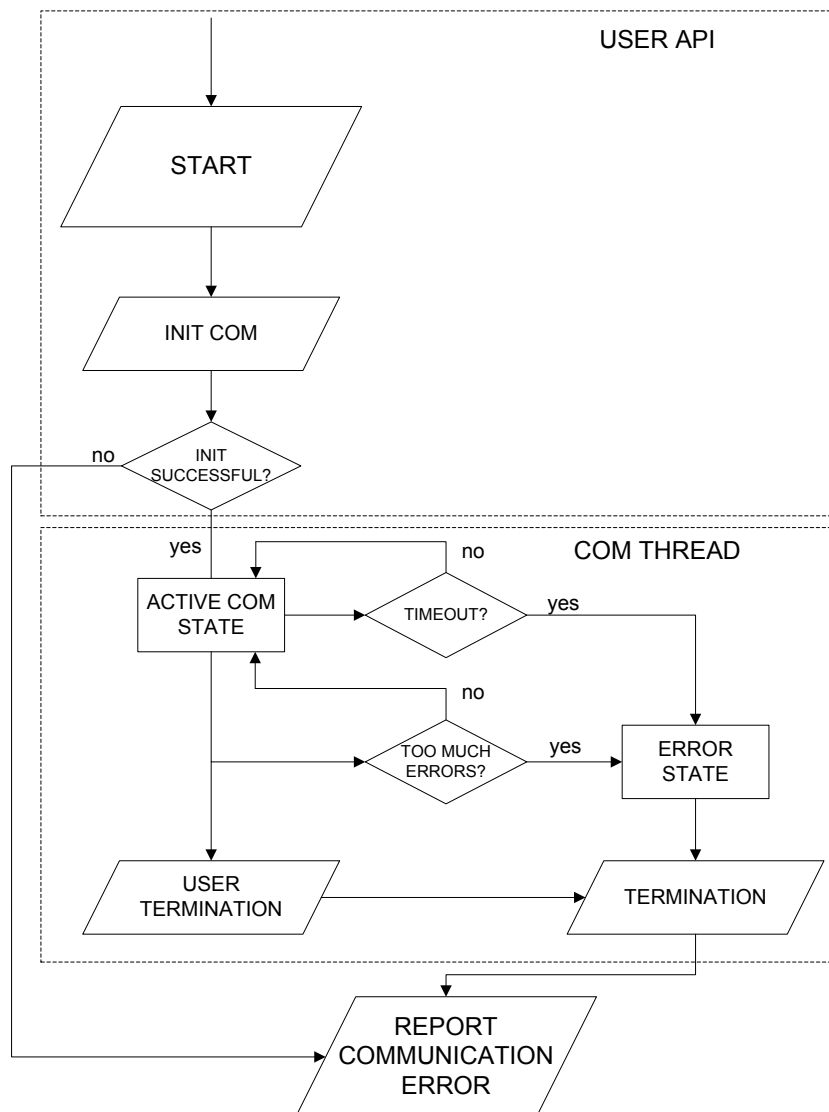


Figura 3.34: Esquema do software que implementa o protocolo de comunicação na entidade externa

trama sem qualquer informação de atuação se o utilizador não tiver dado nenhuma ordem. Observou-se este cuidado para que a plataforma não declare falha de comunicação.

A comunicação pode terminar por três motivos: por acumulação de erros, porque a plataforma deixou de responder, ou porque o utilizador decidiu que esta devia ser terminada. Nos dois primeiros casos a terminação ocorre sincronamente com a execução periódica da *thread* em que é verificado se se acumularam pelo menos 5 erros de descodificação de tramas ou se já se está há mais do que 160 ms sem receber qualquer trama. Qualquer uma destas situações conduz à finalização da execução da *thread* e ao consequente registo de uma situação de erro que será comunicada ao utilizador da próxima vez que este invocar uma função da API de suporte. A finalização da execução por parte do utilizador tem efeitos imediatos sendo a execução da *thread* interrompida.

API de suporte da entidade externa

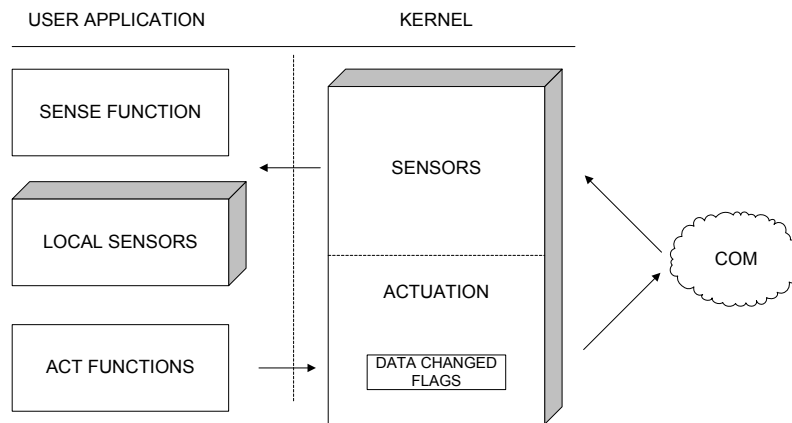


Figura 3.35: Esquema conceptual da API da entidade externa

Na entidade externa foi criada uma API de suporte para que se possam criar aplicações que interajam com a plataforma robótica. Esta API de suporte foi desenhada de forma a ser o mais semelhante possível com aquela que foi implementada na plataforma robótica. Desta forma, o desenvolvimento de aplicações usufrui de uma interface semelhante quer se trate de uma aplicação para ser executada na unidade de controlo da plataforma, quer se trate de uma aplicação para ser executada na entidade externa.

A estrutura desta API e a sua comunicação com a entidade externa pode ser melhor entendida com a observação da Figura 3.35. Nesta figura assinala-se o núcleo da API como (*KERNEL*) do *software* de comunicação. Esta possui uma interface com a aplicação desenvolvida pelo utilizador, assinalada como *USER APP*, que consiste num conjunto de estruturas de dados onde são armazenadas as mais recentes recolhas sensoriais da plataforma e as ordens de atuação que esta deve executar. O utilizador fica encarregue de criar uma base de dados local para a sua aplicação, esta é sincronizada com a base de dados do *KERNEL* mediante o pedido para que tal seja feito através das funções da API a isso destinadas. O utilizador pode, após esta sincronização, usar os dados presentes na sua base de dados local. Esta implementação permite transferir os problemas de exclusão mútua criados por um ambiente *multithread* para a responsabilidade da API que irá gerir o acesso à base de dados do *KERNEL* evitando que

o utilizador se tenha de preocupar com isso.

No que toca à atuação, a API põe ao dispor do utilizador um conjunto de funções que permitem atuar sobre a plataforma. Estas atualizam os dados para atuação presentes na base de dados do *KERNEL*, assinalada como *ACTUATION*. Para saber quais foram os dados atualizados e como se deve atuar sobre a plataforma existe um conjunto de *flags* que marca as alterações. Posteriormente, a *thread* que suporta a comunicação sabe que tramas deve construir e enviar para a plataforma.

Capítulo 4

Resultados

Neste capítulo apresentam-se os resultados obtidos com a realização desta dissertação de mestrado. Alguns dos resultados apresentados já foram referidos nos capítulos anteriores por serem imprescindíveis para o desenvolvimento do texto, estes serão apenas apresentados de forma mais organizada para uma consulta mais fácil.

Este capítulo está dividido em três secções. Na primeira descrevem-se as características da plataforma para que sirvam para referência futura. Na Secção 4.2 referir-se-á um software criado para teste da plataforma. Por fim far-se-á, na Secção 4.3 uma comparação com algumas plataformas existentes no mercado e já referidas.

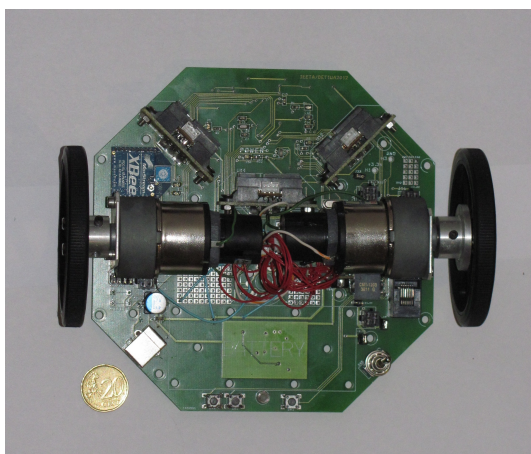
A construção de um protótipo permitiu que se possa obter alguns resultados de uma forma prática, possibilitando a sua comparação com o que se previu nos capítulos anteriores. Este protótipo pode ser observado na Figura 4.1 em várias vistas.

4.1 Caraterização da plataforma

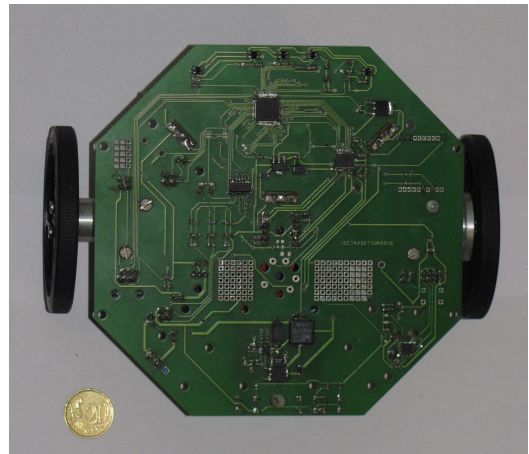
O protótipo construído apresenta um conjunto de características que já se referiram anteriormente, contudo, de forma dispersa. Para referência futura resumem-se na Tabela 4.1 as características que foram sendo apresentadas.

Das características apresentadas na Tabela 4.1, pode comprovar-se praticamente a velocidade máxima uma vez que, para obter este valor teórico, tem-se partido da velocidade máxima em vazio dos motores que é diferente daquela que eles apresentam quando têm alguma carga. Usou-se para isso a API implementada na entidade externa em conjunção com os *encoders* presentes na plataforma. Fizemos dois ensaios, o primeiro dos quais em vazio e o segundo com o robô a deslocar-se ao longo de uma superfície plana. A ponte-H estava a ser alimentada com uma tensão de 10 V, sendo que o PWM estava limitado a 72%. Dos testes efetuados obtiveram-se os gráficos da Figura 4.2.

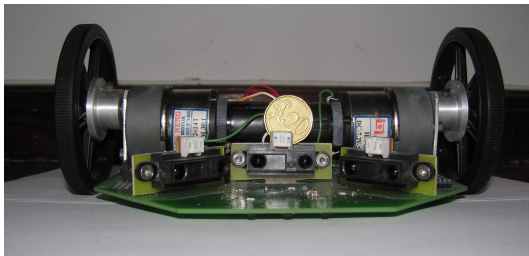
Com os gráficos da Figura 4.2 obtém-se uma caraterização da locomoção do robô que permite concluir mais acerca do sistema de locomoção. Quanto à velocidade máxima da plataforma pode concluir-se que esta ronda os 0,55 m/s com as rodas em vazio e, sendo limitada pela menor das velocidades, das rodas é de 0,51 m/s quando a plataforma se desloca ao longo de uma superfície plana. Esta velocidade é muito inferior àquela que calculámos teoricamente e, dado isto, deve rever-se o *setpoint* máximo que se pode aplicar ao sistema de locomoção. Resolvendo a Expressão 3.2 já referida na Secção 3.3.2 para esta nova velocidade obtém-se:



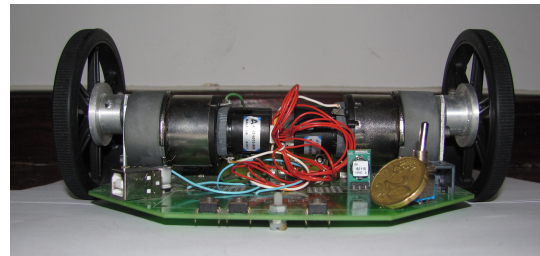
(a) Plataforma robótica, vista de topo



(b) Plataforma robótica, vista de baixo

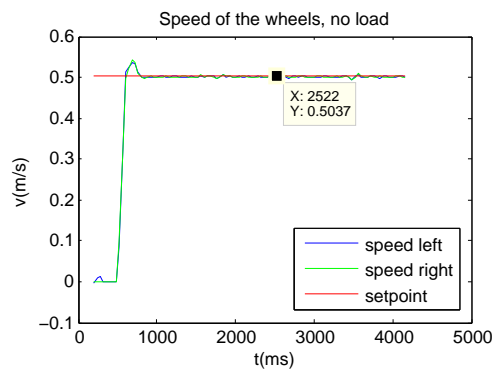


(c) Plataforma robótica, vista de frente

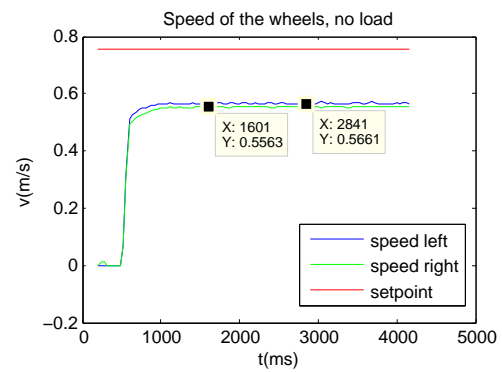


(d) Plataforma robótica, vista de trás

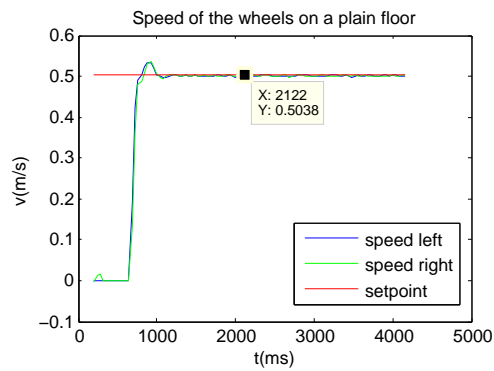
Figura 4.1: Várias vistas do protótipo construído



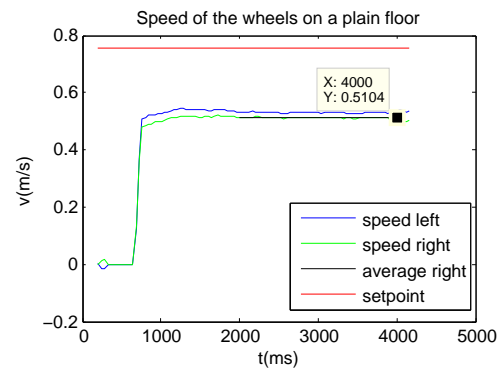
(a) Setpoint 0,50m/s



(b) Setpoint 0,76m/s



(c) Setpoint 0,50m/s



(d) Setpoint 0,76m/s

Figura 4.2: Gráficos da velocidade da plataforma para diferentes *setpoints*

Tabela 4.1: Resumo das características da plataforma decorrentes do seu projeto

Característica	Descrição
Dimensões	$184 \times 159 \times 80\text{mm}^3$
Tensão de alimentação	3 a 15 V
Estimativa de potência máxima	18,6 W
Conjunto de sensores	5 sensores de superfície, 3 sensores de distância, 1 sensor de nível de bateria, 1 sensor de farol, 2 encoders e 3 Botões
Conjunto de atuadores	2 Motores de locomoção, 1 Servo-Motor, 3 LEDs e 1 Buzzer
Unidade de Controlo	PIC32MX795F512H, Arquitetura MIPS de 32 bits com 512KB de memória de programa e 128KB de RAM
Comunicação	1 CAN, 1 I2C, 1 USB (Conversão posterior para UART para programação e comunicação com a unidade de Controlo), 1 Xbee, 1 SPI (Não se pode usar em simultâneo com Xbee), 2 UART (Não se podem usar em simultâneo com SPI, apenas 1 se usarmos Xbee)
Velocidade máxima teórica	1,2 m/s

$$Set = \frac{\frac{0,51}{\pi \times 0,08} \times 960}{99,9} = 19$$

Por observação dos gráficos observa-se também um pequeno *overshoot* da velocidade quando o *setpoint* é de 0,50 m/s o que indica que o ganho integral do compensador PI usado deve ser diminuído afim de mitigar este problema [49].

Os sensores de obstáculos usados merecem também algum esforço de caraterização. A tensão de saída é inversamente proporcional à distância a que um obstáculo se encontra, como foi dito anteriormente. Contudo, não se referiu como se pode, a partir dos valores lidos, obter o valor da distância efetiva da plataforma ao obstáculo. Existem, certamente, diferenças, de sensor para sensor, por isso interessa a caraterização concreta dos sensores usados. Nos gráficos da esquerda da Figura 4.3 podem ver-se os dados obtidos dos sensores de obstáculos quando se varia a distância destes a um objeto branco.

Em [62] é definido um método que permite obter uma linearização das curvas de saída características destes sensores. Este método conduz posteriormente a uma expressão que nos permite calcular a distância a que um obstáculo se encontra a partir do valor lido da ADC. Num primeiro passo é usado um método de tentativa erro para achar o valor de k na Expressão 4.1a que melhor linearize os dados. Segundo os autores, para o caso dos sensores Sharp GP2D12 que são os que estão montados na plataforma o valor de k deve ser 4. Em seguida, para aproximar os dados já linearizados usa-se a Expressão 4.1b. Nestas expressões, V é o valor lido do ADC, R a distância a que o obstáculo se encontra, m e b os parâmetros da regressão linear a calcular, do tipo $y = mx + b$ e k é igual a 4 como já foi referido.

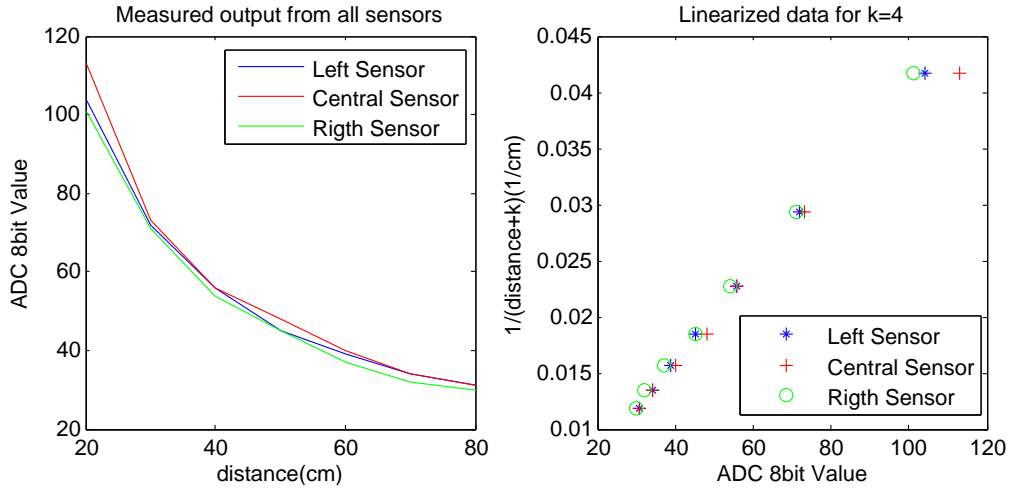


Figura 4.3: Gráficos obtidos com os dados dos sensores de obstáculos e seu processamento

$$\frac{1}{R+k} = m \times V + b \quad (4.1a)$$

$$R = \frac{1}{m \times V + b} - k \quad (4.1b)$$

Aplicando para os três sensores usados, obtiveram-se os valores que se podem ver na Tabela 4.2.

Tabela 4.2: Valores de m e b obtidos para os vários sensores de obstáculos

Sensor	$m(\text{cm}^{-1})$	b
Sensor esquerdo	$0,405 \times 10^{-3}$	$-0,165 \times 10^{-3}$
Sensor central	$0,365 \times 10^{-3}$	$-1,27 \times 10^{-3}$
Sensor direito	$0,413 \times 10^{-3}$	$-0,082 \times 10^{-3}$

Referiu-se anteriormente um consumo de potência máximo estimado de 18,6 W. Interessa saber qual o valor real deste consumo de forma a conhecer as necessidades energéticas da plataforma para isso recorreu-se a um amperímetro e a um voltímetro para conhecer a potência que a plataforma consome. Fiz-se um ensaio com a plataforma a deslocar-se ao máximo da sua velocidade ao longo de uma superfície plana, Para isso usou-se a API implementada na entidade externa, garantindo que todos os sensores estão ligados pois toda a informação sensorial é enviada periodicamente para a entidade externa. Realizou-se, também, outro ensaio com a plataforma no estado *IDLE STATE* descrito quando se referiu a implementação do protocolo de comunicação na plataforma robótica. Para os dois ensaios os valores de potência consumida apresentam-se na Tabela 4.3.

O valor do consumo de potência ficou aquém do valor teórico. Com estes consumos e alimentando a plataforma com uma bateria de 3,7 V de tensão e 2200 mAh consegue-se alimentar a plataforma durante aproximadamente 7 horas em *IDLE STATE* e 1,8 horas com os motores a deslocar-se à velocidade máxima ao longo de uma superfície plana. Considera-se

Tabela 4.3: Potência consumida pela plataforma no ensaios realizados

Ensaio	Potência(W)
Máximo de velocidade em superfície plana	4,5 W
<i>IDLE STATE</i>	1,1 W

uma boa autonomia, embora 1,1 W de consumo em *IDLE STATE* seja um valor um pouco elevado. Deve, contudo considerar-se que, neste estado, ainda se mantêm ligados os *encoders*, a unidade de controlo, o Módulo *XBee* e que os próprios reguladores de tensão consomem alguma corrente para se manterem ligados.

Interessa também dedicar algumas palavras à comunicação *wireless* por implementada e que não mostrou um funcionamento fiável, testes mais concretos não foram conduzidos para se obter uma caracterização objetiva desta funcionalidade da plataforma. Contudo, pode descrever-se o problema: em primeiro lugar a comunicação deixa de funcionar a partir de uma distância superior a aproximadamente 1 metro da plataforma e, em segundo lugar, mesmo para distâncias inferiores deixa de funcionar quando os motores de locomoção são ligados. Pensa-se que a proximidade do módulo *XBee* aos motores de locomoção possa estar a degradar a sua performance, contudo, não havia outro local disponível para a colocar. Sugere-se que sejam efetuados testes mais objetivos para testar esta tese e que, caso se confirme, esta seja mudada para uma das possíveis expansões.

Para finalizar esta caracterização da plataforma construída interessa apresentar o seu peso. Recorrendo a uma balança chegou-se ao valor de 476 g. Deve ter-se em conta que este é o peso do protótipo construído o qual não tem acoplado o sistema de deteção do farol nem qualquer bateria.

4.2 *Software* para teste da plataforma

Para testarmos os vários componentes da plataforma construímos, com a ajuda da API implementada na entidade externa, um *software* que nos permite testar os vários sensores e atuadores presentes na plataforma e efetuar algumas experiências comportamentais. Estas permitem perceber a integração dos vários sistemas da plataforma facilitando, em primeiro lugar, a avaliação do comportamento dos vários sistemas, e, em segundo lugar, a avaliação do comportamento da plataforma como um todo. Construímos uma *framework* de teste que se baseia num menu que é apresentado ao utilizador com diversas opções que este pode escolher.

A interface com esta *framework* faz-se por meio do menu que podemos ver a seguir:

+=====+		
	Microrato testing framework	
	+=====+	
	0 - EXIT	
	+-----+	
	2 - Read obstacle sensors	3 - Read line sensors
	4 - Read IV sensor	5 - Read battery sensors
	6 - Read odometry	7 - Read buttons
	+-----+	
	8 - Set speed	9 - Set servo position
	10 - Beep buzzer	11 - Set/reset leds
	+-----+	
	12 - Magnet robot	13 - Anti-Magnet Robot
	+=====+	

As primeiras opções, desde a 2 até à 7, dizem respeito a leituras de sensores. Já as opções entre 8 e 11 dizem respeito a atuação sobre a plataforma. As duas últimas opções dizem respeito a testes comportamentais com a integração de vários subsistemas da plataforma robótica.

Nesta versão da *framework* construída, apenas se incluíram duas experiências comportamentais que permitem integrar o sensor de obstáculos central e o sistema de locomoção. Na primeira destas experiências o robô é atraído por um obstáculo e na segunda repellido.

4.3 Comparação com plataformas semelhantes

No Capítulo 2 foram abordadas várias plataformas existentes no mercado que, neste ponto, podemos comparar com aquela que desenvolvemos. Resumimos, então, na Tabela 4.4 as características mais importantes das várias plataformas.

O preço apresentado para a plataforma construída pode ser encontrado no Anexo 7 onde se faz a sua estimativa.

Da análise da Tabela 4.4 podemos concluir várias coisas, em primeiro lugar podemos concluir que a plataforma por nós desenvolvida é a que apresenta um maior custo quando comparamos a nossa estimativa com o custo de mercado das outras plataformas apresentadas. Devemos, antes de mais, ter em conta que quando realizámos a estimativa de custo partimos apenas de uma previsão de produção de 5 unidades, enquanto que os preços de mercado apresentados referem-se a plataformas produzidas em série. Uma vez que foi referido que se pretendia construir uma plataforma robótica de baixo custo olhando para a tabela apresentada podemos ver que, comparando com plataformas semelhantes, o custo desta não é muito apelativo. Contudo, se repararmos também nas características, podemos ver que a nossa plataforma apresenta um conjunto de sensores mais alargado, ao mesmo tempo que, também apresenta um intervalo maior para a tensão de entrada possibilitando maior versatilidade em termos de escolha da bateria. Aliado a isto vem também o facto de a unidade de controlo que se usou ser um microcontrolador que abre bastantes possibilidades. Tem bastantes recursos disponíveis e a própria construção da plataforma possibilita que futuras expansões possam comunicar diretamente com a unidade de controlo.

Mesmo assim, se quisermos reduzir os custos da plataforma podemos sempre optar por não assemblar algumas partes que consideremos supérfluas. Podemos, por exemplo, decidir

Tabela 4.4: Comparação da plataforma desenvolvida com as plataformas disponíveis no mercado

Característica	Plataforma desenvolvida	Pololu 3 π [40]	EvalBot [5]	Arduino Robot [41]
Tensão de alimentação	3 a 15 V	3 a 7 V	3,5 a 5 V	4,8 a 5,6 V
Conjunto de sensores	5 sensores de superfície, 3 sensores de distância, 1 sensor de nível de bateria, 1 sensor de farol, 2 encoders e 3 Botões	5 sensores de superfície, 3 botões e um potenciometro	Sensores de toque, encoders e 2 botões	5 Sensores de superfície, Bússula, 5 Botões e potenciometro
Conjunto de atuadores	2 Motores de locomoção, 1 Servo-Motor, 3 LEDs e 1 Buzzer	2 Motores de locomoção 2 LEDs, 1 Buzzer e 1 display LCD	2 Motores de locomoção 2 LEDs, 1 Speaker e 1 display OLED	2 Motores de Locomoção, LEDs e display LCD a cores
Unidade de Controlo	PIC32MX795, Arquitetura MIPS de 32 bits com 512 KB de memória Flash e 128 KB de RAM	ATmega 168/328P, arquitetura AVR de 8 bits com, respetivamente, 16/32 KB de memória Flash e 1/2 KB de memória SRAM	Stellaris LM3S9B92, arquitetura ARM de 32bits 256 KB de memória Flash e 96 KB de memória SRAM.	2 x ATmega32u4, arquitetura AVR de 8 bits com 32 KB de memória Flash e 2,5 KB de memória SRAM. Memória adicional para um dos Microcontroladores
Comunicação	1 CAN, 1 I2C, 1 USB, 1 Xbee, 1 SPI, 2 UART	Apenas alguns portos de I/O e linhas analógicas disponíveis	USB host e device e Ethernet	1 UART, 1 USB, 1 SPI e 1 I2C
preço	294,18 EUR	84,95 EUR	121,65 EUR	189 EUR

que vamos alimentar a plataforma com uma tensão superior a 10 V e abdicar do regulador *step up*. O facto de a PCB que constitui a base da plataforma robótica ter sido projetada a contar com todos os componentes que temos vindo a referir não quer dizer que não possamos abdicar de alguns deles, baixando, assim, o custo da plataforma. Parte-se do princípio que, a montagem da plataforma vai ser efetuada por alguém capaz de discernir do que é que pode abdicar sem prejudicar a performance da mesma.

Capítulo 5

Conclusões e trabalho futuro

Nesta dissertação de mestrado descreve-se o processo de conceção, projeto e construção de uma plataforma robótica didática. Iniciou-se o processo de conceção com uma descrição acerca da robótica educacional, a forma como esta entra nos currículos educacionais e as suas potencialidades. As plataformas robóticas tornam-se importantes pois são um veículo didático neste contexto. De várias plataformas robóticas apresentadas tomou-se inspiração para a construção de uma plataforma pequena e fácil de assemblar, para isso contribuiu o facto de se ter reduzido a estrutura da plataforma a uma PCB onde assenta toda a sua eletrónica e mecânica.

A plataforma robótica foi projetada de forma a contemplar um conjunto básico de sensores e atuadores que lhe permitissem a participação no concurso Micro-Rato organizado pela UA desde 1995. Embora o projeto estivesse em muito relacionado com este concurso, as suas aplicações não se restringem apenas a ele. É uma plataforma que pode ser usada nos mais diversos contextos. A inclusão nesta de um microcontrolador de 32 bits com bastantes recursos disponíveis, quer em termos de memória quer em termos de periféricos, permite implementar algoritmos de controlo mais complexos na própria plataforma. Os sensores de superfície usados foram colocados de forma a potenciar aplicações em que é possível o seguimento de uma linha, podendo, inclusivamente detetar-se cruzamentos de linhas, o que pode potenciar aplicações como a resolução de labirintos definidos por meio de linhas, o que é, agora, uma competição incluída do concurso Micro-Rato. Também a inclusão de *encoders* permitindo através de algoritmos de odometria conhecer a posição da plataforma em relação ao ponto de partida potencia novas aplicações. Acresce a tudo isto, o ambiente de teleoperação criado em que se pode, numa entidade externa com um poder de processamento superior ao da plataforma robótica, construir aplicações que, conhecendo os dados sensoriais da plataforma, os possam trabalhar com maior poder computacional. As aplicações construídas na entidade externa podem também atuar sobre esta.

As escolhas que se fizeram dos vários componentes a incluir na plataforma mostraram-se, na sua maioria, adequadas à aplicação em questão. O sistema de locomoção, embora mais lento que o teoricamente previsto, apresenta uma velocidade máxima de 0,51 m/s sendo esta controlada em malha fechada através dos *encoders* acoplados aos motores de locomoção. Isto significa que a plataforma construída é consideravelmente rápida e, mitigando possíveis imperfeições dos motores de locomoção através do controlo em malha fechada, mantém-se em linha reta para velocidade iguais aplicadas a ambos os motores, o que poderia não se verificar com um controlo em malha aberta. Os sensores de obstáculos escolhidos demonstraram,

praticamente, a possibilidade de converter o valor lido das suas saídas na distância a que um obstáculo se encontra.

Era um objetivo desta dissertação que houvesse um forma de comunicação *wireless* com a entidade externa. A implementação efetuada não demonstrou um comportamento adequado para a aplicação em questão, deixando de funcionar a distâncias superiores a um metro.

Em resumo, em relação às plataformas existentes no mercado que foram apresentadas, esta apresenta, um conjunto de sensores maior e a vantagem de possuir uma unidade de controlo com as vantagens já apresentadas. A existência de um ambiente de teleoperação implementado numa entidade externa confere também uma vantagem à plataforma construída em relação às existentes no mercado.

Para trabalho futuro deve implementar-se outra solução de comunicação, ou melhorar a existente, de forma a que a comunicação *wireless* se torne uma mais valia desta plataforma.

É aberto um grande leque de possibilidades com esta plataforma que deve ser aproveitado no futuro para aplicações de robótica educacional. Será útil para o ensino de princípios de robótica e até de conteúdos de outras áreas do saber o uso desta plataforma como veículo de disseminação de conhecimento, permitindo um largo conjunto de experiências práticas. Para implementação em *software* destas experiências pode usar-se, tanto a unidade de controlo da plataforma, como o ambiente de teleoperação criado na entidade externa.

Bibliografia

- [1] Carnegie Mellon University. Mobot races, 2012. [Acedido: 24-10-2012]. URL: <http://www.cs.cmu.edu/mobot/>.
- [2] Robotic Arena. Robotic arena, 2012. [Acedido: 25-10-2012]. URL: <http://lirec.ict.pwr.wroc.pl/~arena/>.
- [3] Micro-Rato. *Regras e especificações técnicas da modalidade Micro-Rato*, 2007.
- [4] Milton Ruas and José Luís Azevedo. Robot voyager ii - reactividade e eficiência. *Revista do DETUA*, Vol. 4, Nr. 5, 2005.
- [5] Texas Instruments. Ek-evalbot evaluation kits. [Acedido: 1-10-2013]. URL: <http://www.ti.com/tool/ek-evalbot>.
- [6] AEAT-6010/6012 magnetic encoder, 10 or 12 bit angular detection device. URL: www.avagotech.co.jp/docs/AV02-0188EN.
- [7] Servocity. How do servos work. [Acedido: 12-10-2013]. URL: http://www.servocity.com/html/how_do_servos_work_.html.
- [8] Roboshop. Compare infrared and ultrasonic distance sensors, 2012. [Acedido: 7-11-2012]. URL: <http://www.robotshop.com/compare-infrared-ultrasonic-distance-sensors.html>.
- [9] Lynxmotion GHM-03 pmc motors. URL: <http://www.lynxmotion.com/images/data/ghm03.pdf>.
- [10] Hitec general servo information. URL: <http://www.hitecrcd.com/files/Servomanual.pdf>.
- [11] Patrice Oehen. Zigbee: An overview of the upcoming standard, 2013. URL: <http://wenku.baidu.com/view/e1f3c4b91a37f111f1855ba5.html>.
- [12] SENA. The comparison of wi-fi, bluetooth and zigbee, 2010. [Acedido: 17-04-2013]. URL: <http://www.sena.com/blog/?p=359>.
- [13] Wikipédia. Hefesto — wikipédia, a enciclopédia livre, 2012. [Acedido: 25-10-2012]. URL: <http://pt.wikipedia.org/w/index.php?title=Hefesto&oldid=32069045>.
- [14] Wikipedia. Antikythera mechanism — wikipedia, the free encyclopedia, 2012. [Acedido: 25-10-2012]. URL: http://en.wikipedia.org/w/index.php?title=Antikythera_mechanism&oldid=519811706.

- [15] Wikipedia. R.u.r. — wikipedia, the free encyclopedia, 2012. [Acedido: 25-10-2012]. URL: <http://en.wikipedia.org/w/index.php?title=R.U.R.&oldid=518253312>.
- [16] Phillip McKerrow. *Introduction to robotics / Phillip John McKerrow*. Addison-Wesley Pub. Co., Sydney ; Reading, Mass. :, 1991.
- [17] Johan Kummeneje. *RoboCup as a Means to Research, Education, and Dissemination*. PhD thesis, 2003.
- [18] RoboCup. Robocup, 2012. [Acedido: 20-09-2012]. URL: <http://www.robocup.org>.
- [19] Randall D. Beer, Hillel J. Chiel, and Richard F. Drushel. Using autonomous robotics to teach science and engineering. *Commun. ACM*, 42(6):85–92, June 1999. URL: <http://doi.acm.org/10.1145/303849.303866>, doi:10.1145/303849.303866.
- [20] Jacek Malec. Some thoughts on robotics for education. *Presented at the 2001 AAAI Spring Symposium on Robotics and Education, Stanford University*, March 2001.
- [21] Escola Profissional de Braga. Epb no festival nacional de robótica 2012.
- [22] Festival Nacional Robótica. Festival nacional robótica 2012, 2012. [Acedido: 27-09-2012]. URL: <http://www.robotica2012.org>.
- [23] 6.270 Staff. About 6.270, 2010. [Acedido: 27-09-2012]. URL: <http://scripts.mit.edu/~6.270/about/>.
- [24] Robots.net. Robot competitions, 2012. [Acedido: 11-10-2012]. URL: <http://robots.net/rcfaq.html>.
- [25] The Trinity College Robotics Competition Staff. Fire fighting home robot contest, 2012. [Acedido: 13-10-2012]. URL: <http://www.trincoll.edu/events/robot/default.asp>.
- [26] Concurso Robô Bombeiro. Robô bombeiro, concurso nacional de robótica do instituto politécnico da guarda, 2012. [Acedido: 13-10-2012]. URL: <http://robobombeiro.ipg.pt/>.
- [27] Feng-Hsiung Hsu. *Behind Deep Blue: Building the Computer That Defeated the World Chess Champion*. Princeton University Press, Princeton, NJ, USA, 2002.
- [28] RoboCup. Robocup junior, 2012. [Acedido: 19-10-2012]. URL: <http://rcj.robocup.org>.
- [29] MicroMouseOnline. Micromouseonline, 2010. [Acedido: 25-10-2012]. URL: <http://www.micromouseonline.com/>.
- [30] Technology Innovation Centre. Uk micromouse site, 2011. [Acedido: 25-10-2012]. URL: <http://www.tic.ac.uk/micromouse>.
- [31] Evan Ackerman. Meet the new world's fastest micromouse robot, 2011. [Acedido: 25-10-2012]. URL: <http://spectrum.ieee.org/automaton/robotics/diy/meet-the-new-worlds-fastest-micromouse>.

- [32] New Techology Foundation. Micromouse2012, 2012. [Acedido: 25-10-2012]. URL: <http://www.ntf.or.jp/mouse/micromouse2012>.
- [33] Robotic Tournament. Robotic tournament, 2012. [Acedido: 25-10-2012]. URL: <http://www.robotictournament.pl/>.
- [34] Universidade de Trás-os-montes e Alto Douro. Micromouse portuguese contest. [Acedido: 2-11-2013]. URL: <http://www.micromouse.utad.pt/>.
- [35] Micro-Rato. Concurso micro-rato, 2012. [Acedido: 26-10-2012]. URL: <http://microrato.ua.pt/>.
- [36] *CiberRato 2011, Rules and Technical Specifications*.
- [37] Universidade de Aveiro. Fabrica centro ciência viva aveiro, 2012. [Acedido: 26-10-2012]. URL: <http://www.ua.pt/fabrica/>.
- [38] G.A. Bekey. *Autonomous Robots: From Biological Inspiration to Implementation and Control*. Intelligent Robotics and Autonomous Agents Series. MIT Press, 2005. URL: <http://books.google.pt/books?id=3xwfia2DpmoC>.
- [39] Bruno Pires. Plataforma robótica multi-funcional. Dissertação de mestrado, Universidade de Aveiro, 2008.
- [40] Pololu Corporation. Pololu robotics and electronics, 2012. [Acedido: 31-10-2012]. URL: <http://www.pololu.com>.
- [41] Arduino. Arduino robot. [Acedido: 3-10-2013]. URL: <http://arduino.cc/en/Main/Robot>.
- [42] J.L. Jones and A.M. Flynn. *Mobile robots: inspiration to implementation*. A.K. Peters, 1993. URL: <http://books.google.ca/books?id=ijlSAAAAMAAJ>.
- [43] GP2D12, distance measuring sensor unit. URL: http://www.sharpsma.com/webfm_send/1203.
- [44] GP2Y0A21YK0F, distance measuring sensor unit. URL: http://www.sharpsma.com/webfm_send/1208.
- [45] QRE1113, minature reflective object sensor. URL: <http://www.fairchildsemi.com/ds/QR/QRE1113.pdf>.
- [46] QRE1113, minature reflective object sensor. URL: http://www.pololu.com/docs/pdf/0J13/QTR_application_note.pdf.
- [47] GP1UE12, low voltage operation, compact ir detecting unit for remote control. URL: <http://media.digikey.com/pdf/Data%20Sheets/Sharp%20PDFs/GP1UE261X,267X,26X.pdf>.
- [48] TB6612FNG, driver ic for dual dc motor. URL: http://www.toshiba-components.com/docs/linear/TB6612FNG_en_datasheet.pdf.
- [49] A.P. Melo. *Teoria dos Sistemas de Controlo Lineares*. Universidade de Aveiro, 2010.

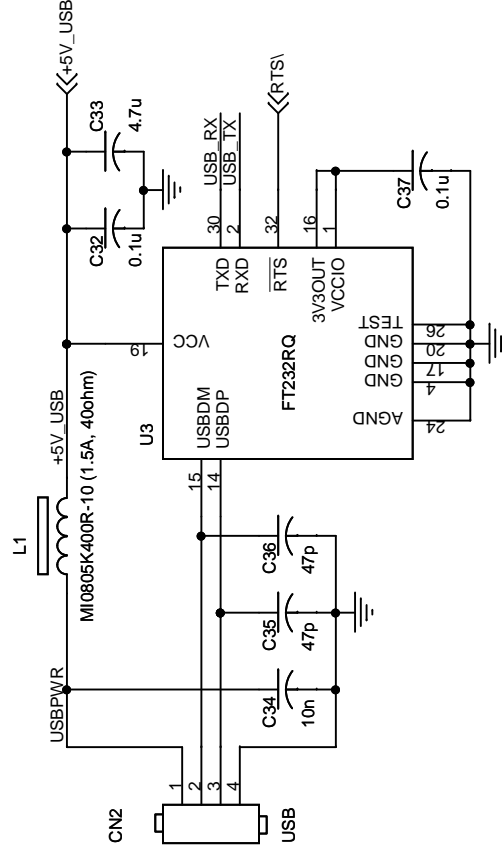
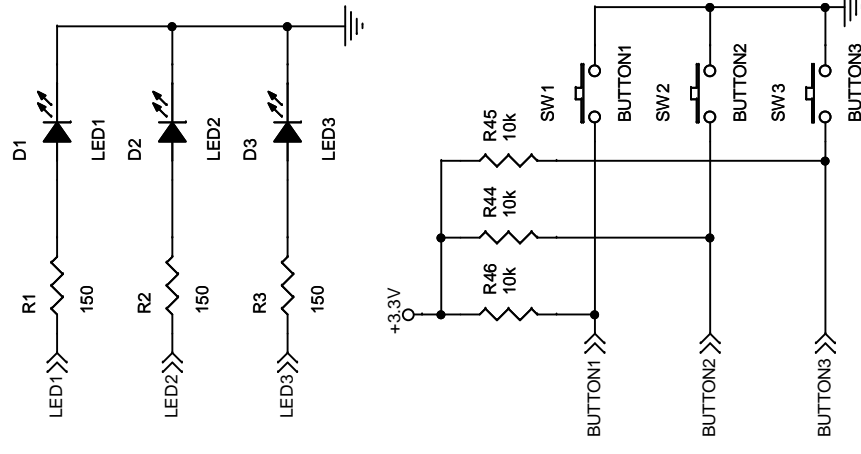
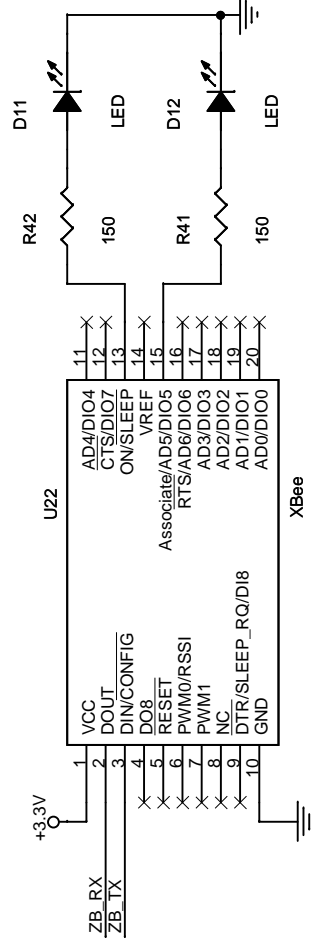
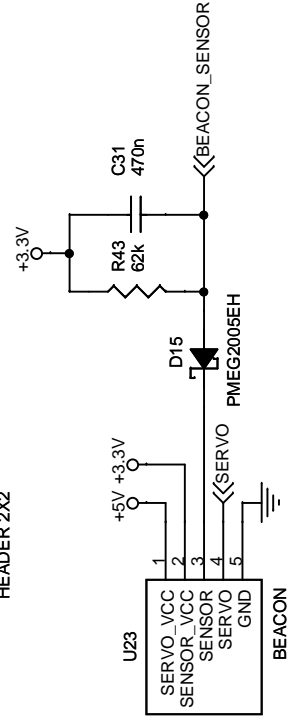
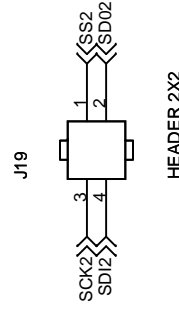
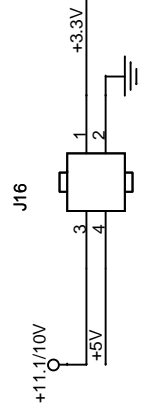
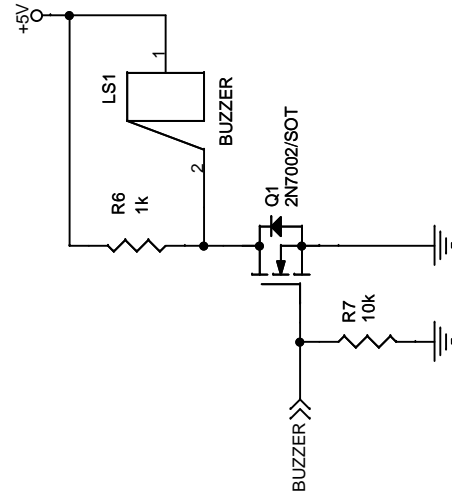
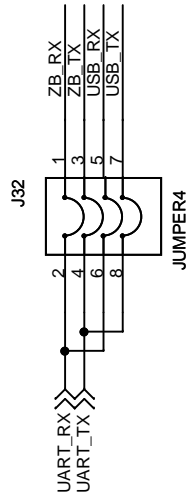
- [50] Society of Robots. Actuators - servos, 2013. [Acedido 25-03-2013]. URL: http://www.societyofrobots.com/actuators_servos.shtml#digitalanalogservos.
- [51] FT232R usb uart ic datasheet version 2.09. URL: http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT232R.pdf.
- [52] Xbee/xbee-pro rf modules. URL: <http://docs-europe.electrocomponents.com/webdocs/0b02/0900766b80b02569.pdf>.
- [53] Microchip. 32-bit pic microcontrollers, 2013. [Acedido 1-06-2013]. URL: <http://www.microchip.com/pagehandler/en-us/family/32bit/architecture.html>.
- [54] Microchip. Pic32mx5xx/6xx/7xx. URL: <http://ww1.microchip.com/downloads/en/DeviceDoc/61156H.pdf>.
- [55] Battery University. Charging lithium-ion. [Acedido em 24-11-2013]. URL: http://batteryuniversity.com/learn/article/charging_lithium_ion_batteries.
- [56] Sam Davis. Schottky diodes: the old ones are good, the new ones are better, 2011. URL: <http://powerelectronics.com/site-files/powerelectronics.com/files/archive/powerelectronics.com/images/SchottkyDiodes.pdf>.
- [57] ServoCity. Hs-5245mg digital mini. [Acedido: 9-06-2013]. URL: http://www.servocity.com/html/hs-5245mg_digital_mini.html.
- [58] Microchip. Tc1262, 500ma fixed output cmos ldo. URL: ww1.microchip.com/downloads/en/devicedoc/21373b.pdf?
- [59] Murata. Oki-78sr series, fixed output 1.5 amp sip dc/dc converters. URL: www.murata-ps.com/data/power/oki-78sr.pdf??
- [60] Maxim. Max668/max669, 1.8v to 28v input, pwm step-up controllers in umax. URL: <http://datasheets.maximintegrated.com/en/ds/MAX668-MAX669.pdf>.
- [61] Maxim. Ee-sim design generation and simulation tool. [Acedido: 9-06-2013]. URL: http://www.maximintegrated.com/ee_sim/.
- [62] Acroname. Linearizing sharp ranger data. [Acedido: 8-10-2013]. URL: <http://www.acroname.com/robotics/info/articles/irlinear/irlinear.html>.
- [63] Eurocircuits. Eurocircuits. [Acedido: 18-06-2013]. URL: <http://www.eurocircuits.com>.

Anexo 1

Esquemático e *layout* da PCB principal

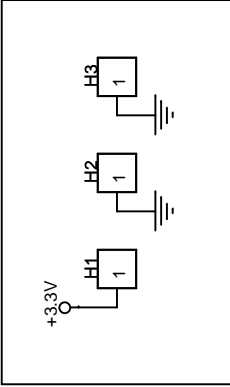
Neste anexo é apresentado o esquemático e *layout* da PCB que constitui a base da plataforma robótica construída.

Nas próximas 5 páginas é mostrado o esquemático e nas 4 seguintes é apresentado o *layout* pela seguinte ordem de *layers*: *Silk Screen Top*, *Top*, *Bottom* e *Silk Screen Bottom*.

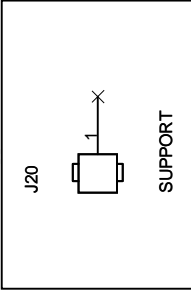


Title		Micro Ratio - Interface	
Size	A4	Document Number	<Doc>
Date:	Tuesday, August 13, 2013	Sheet	1 of 5

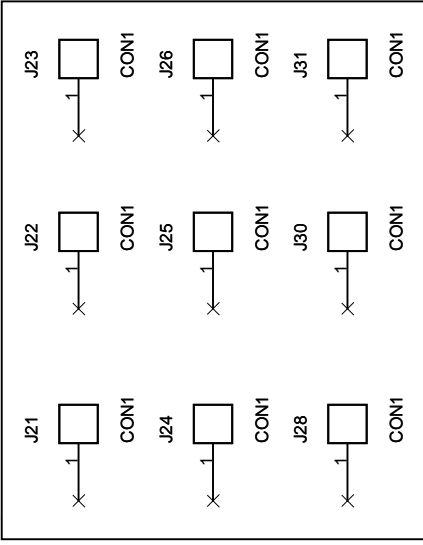
Test Points



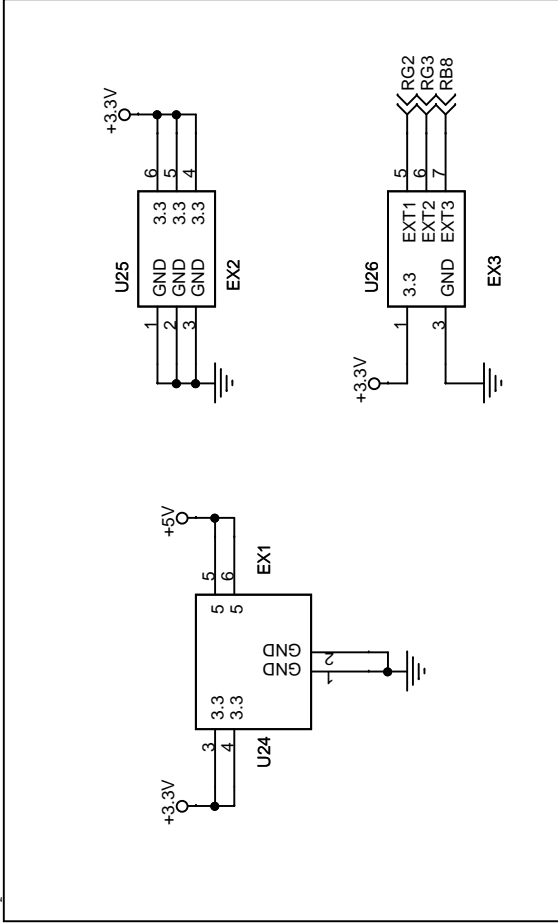
Mechanical Support



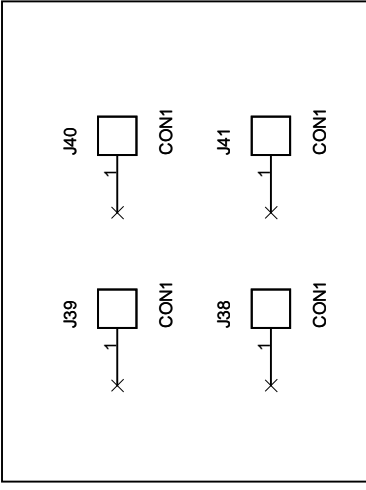
Expansion Holes



Expansion Areas



Motor



Title

Micro Rato - Mechanical

Size

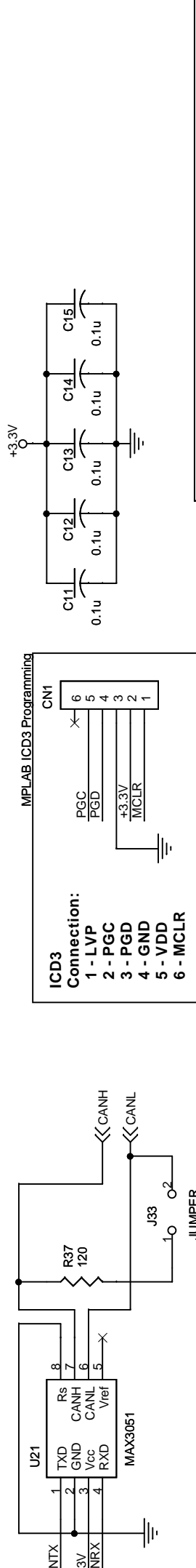
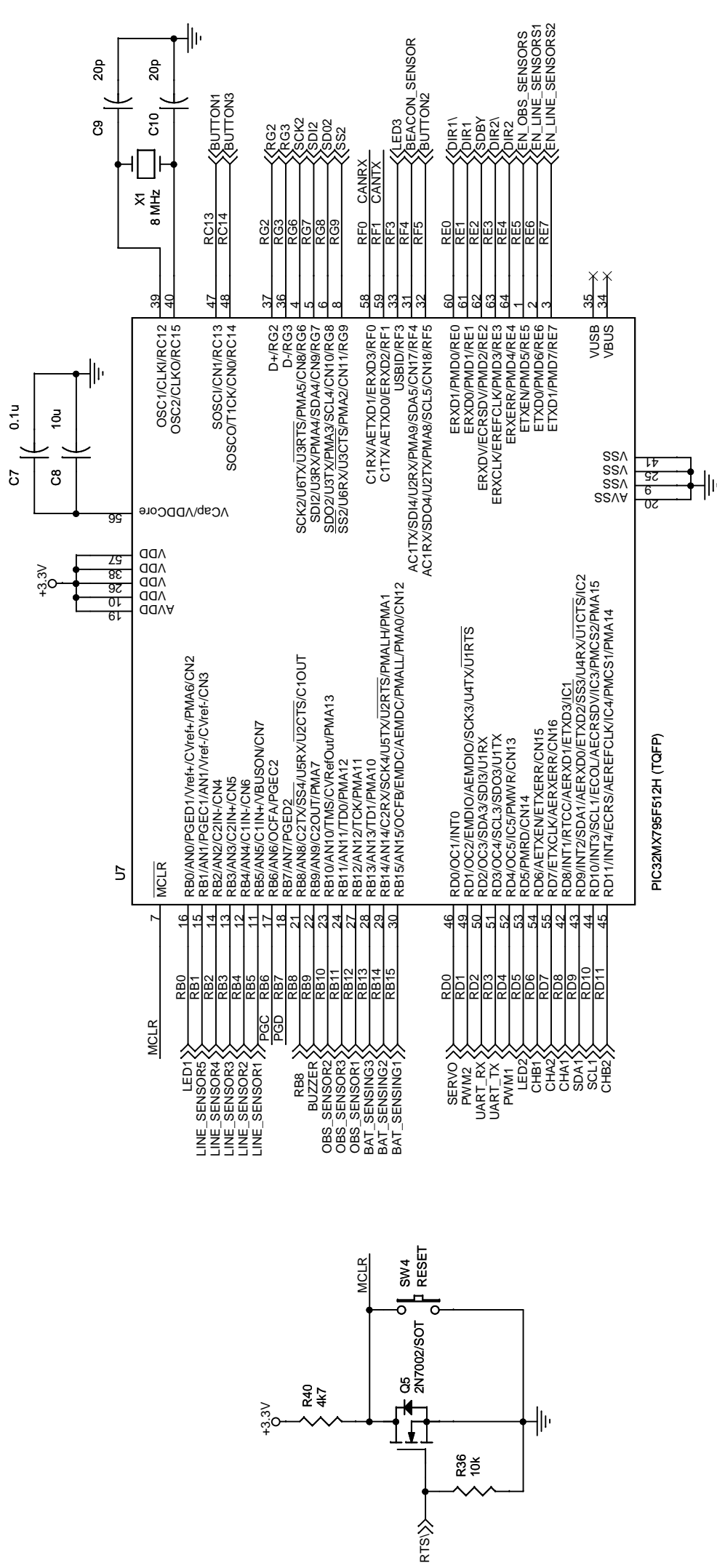
A4

Document Number

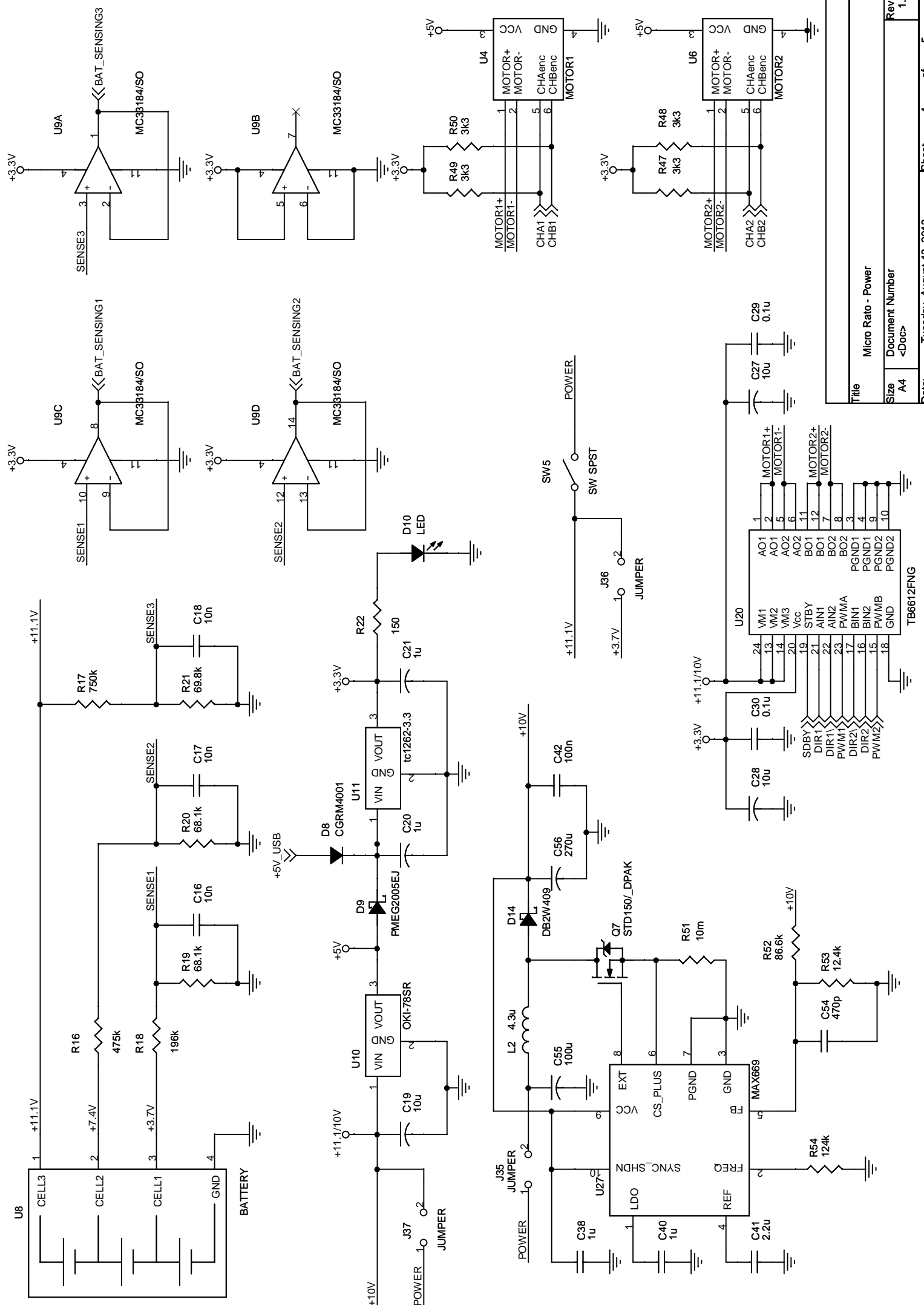
<Doc>

Rev

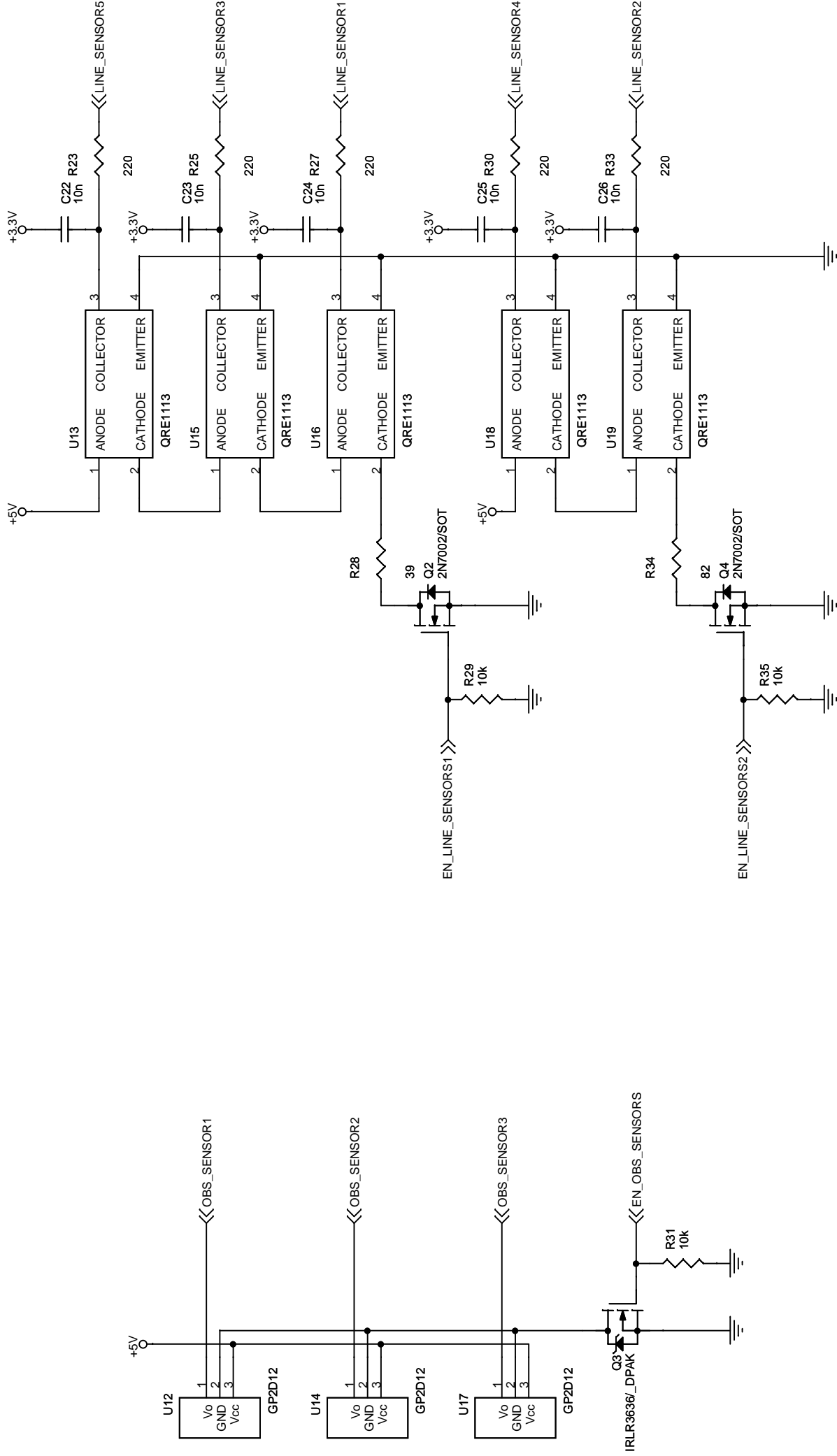
1.0



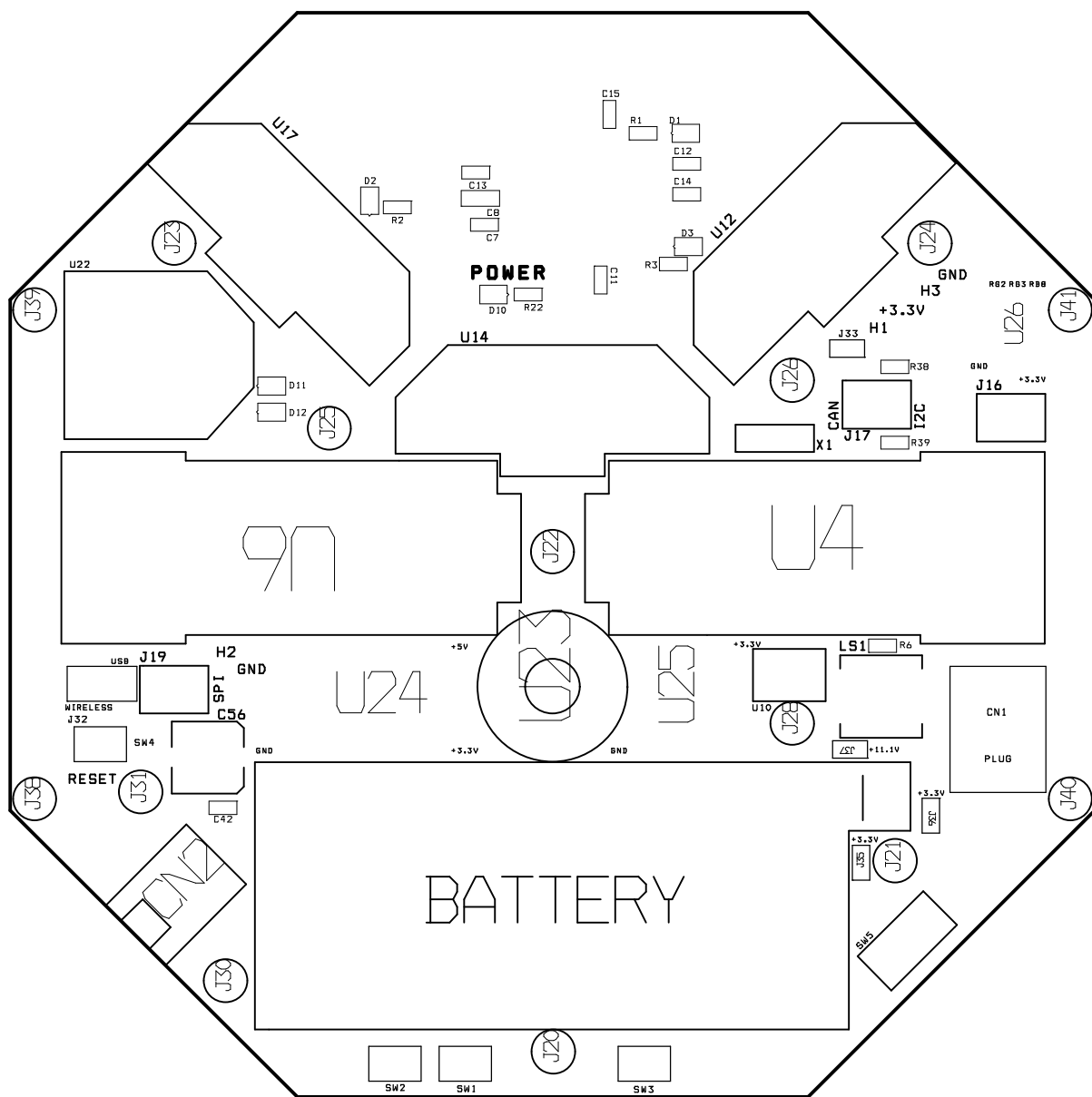
Title				
Micro Ratio - Control				
Size A4	Document Number <Doc>			Rev 1.0
Date:	Tuesday, August 13, 2013	Sheet	3	of 5

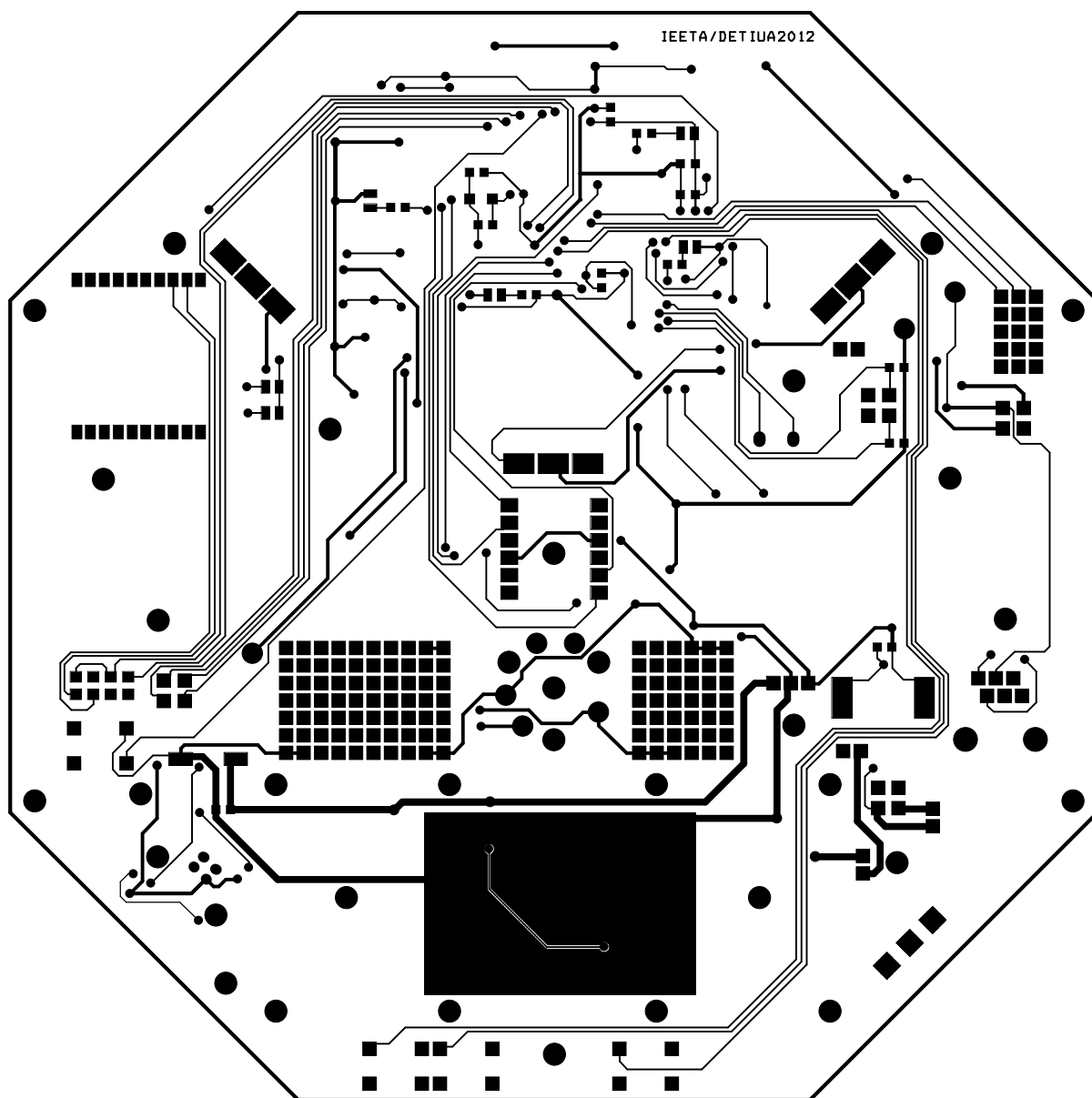


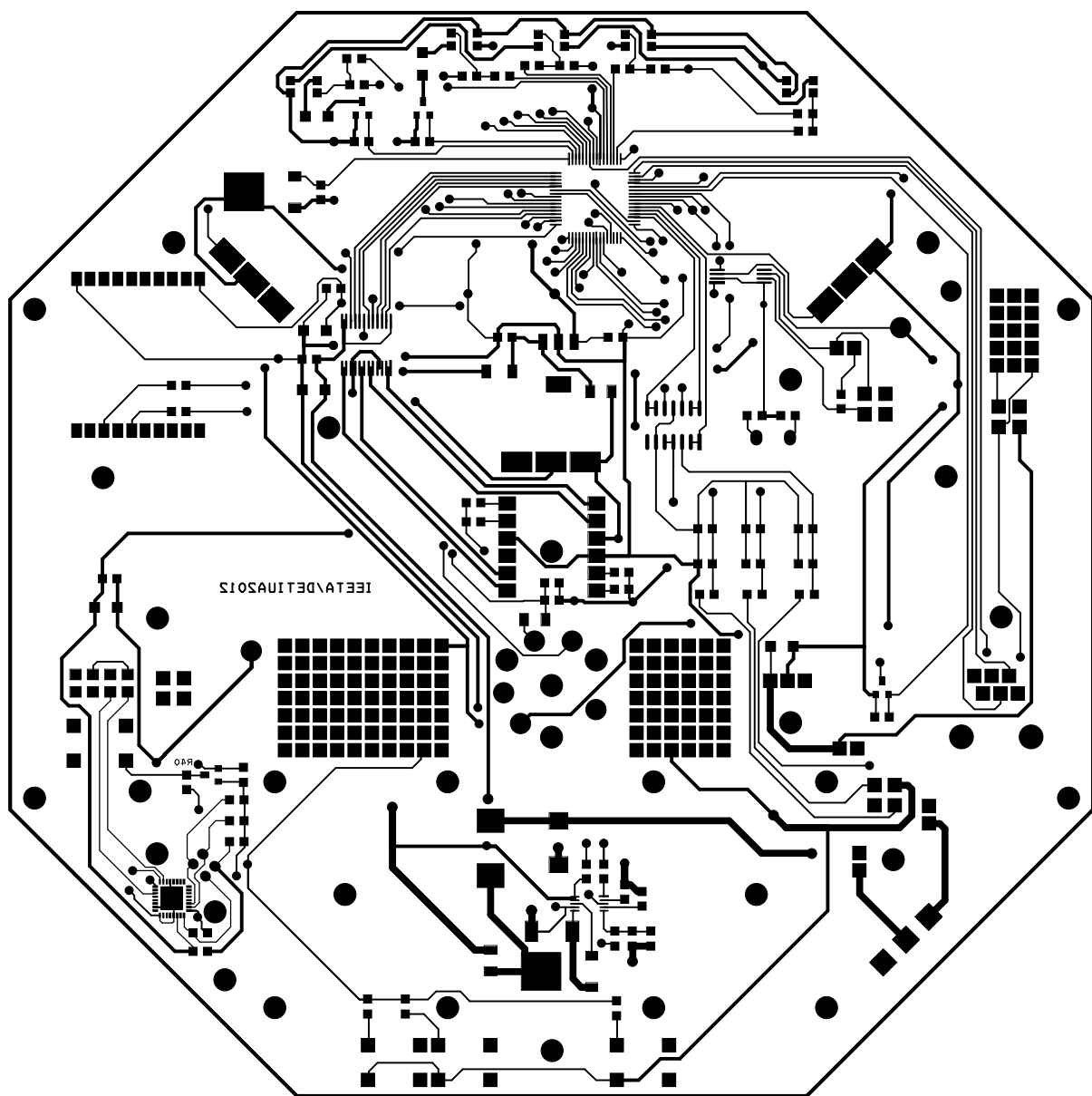
Title		Micro Ratio - Power	
Size	A4	Document Number	<Doc>
Date:	Tuesday, August 13, 2013	Sheet	4 of 5
Rev	1.0		

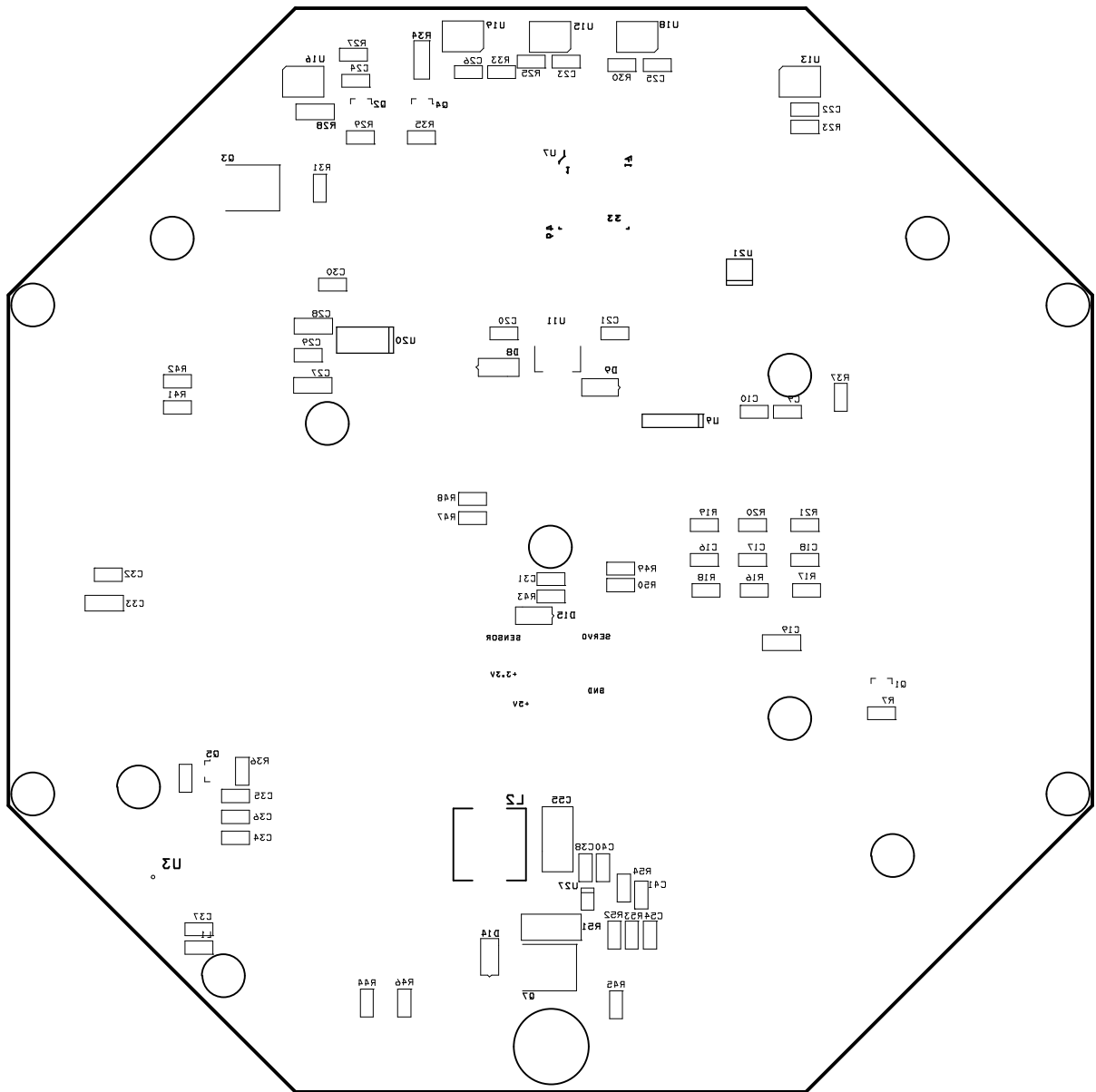


Title		Micro Rato - Sensors		
Size	Document Number	Rev		
A4	<Doc>	1.0		
Date:	Monday, June 04, 2012	Sheet	5	of 5







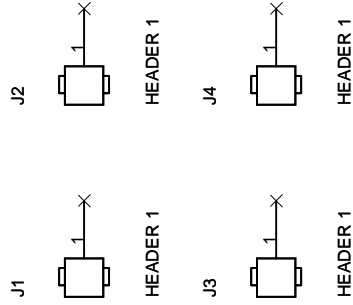
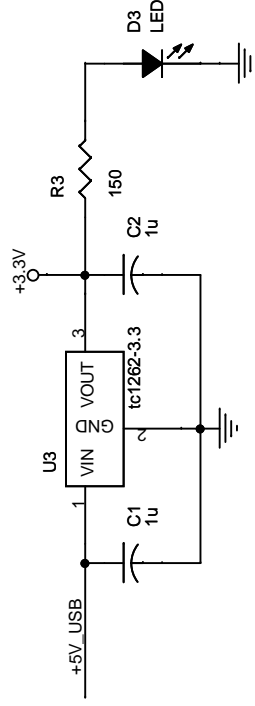
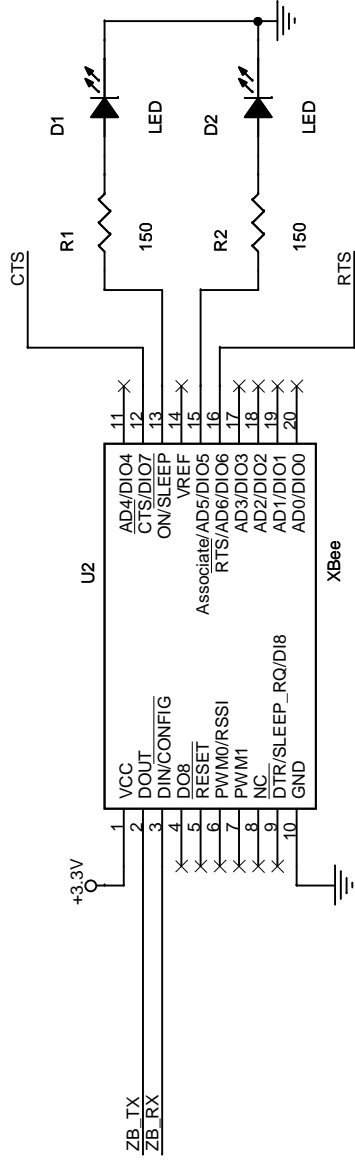
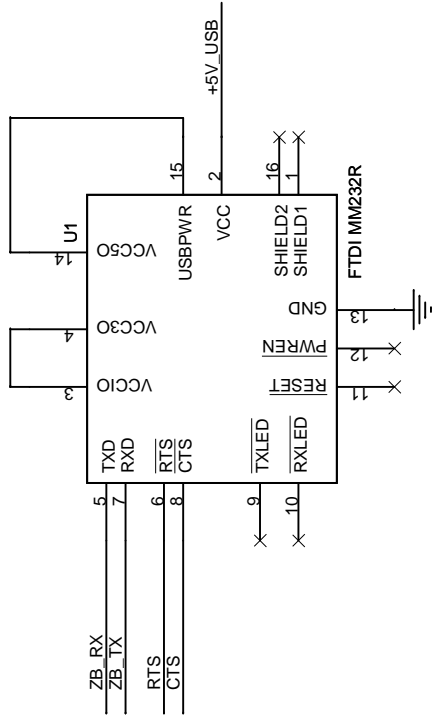


Anexo 2

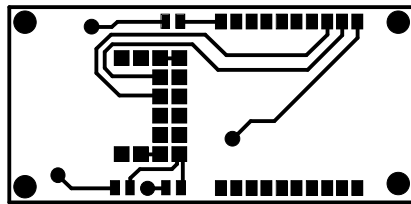
Esquemático e *layout* da PCB do módulo *XBee*

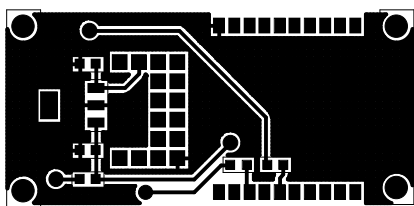
Neste anexo é apresentado o esquemático e *layout* da PCB que foi construída para implementar a comunicação ZigBee na entidade externa, através de um módulo *Xbee*.

Nas próxima página é mostrado o esquemático e nas 2 seguintes é apresentado o *layout* pela seguinte ordem de *layers*: *Top* e *Bottom*.



Title		USB XBee Adapter	
Size	A4	Document Number	<Doc>
Rev	1.0		
Date:	Friday, August 31, 2012	Sheet	1 of 1



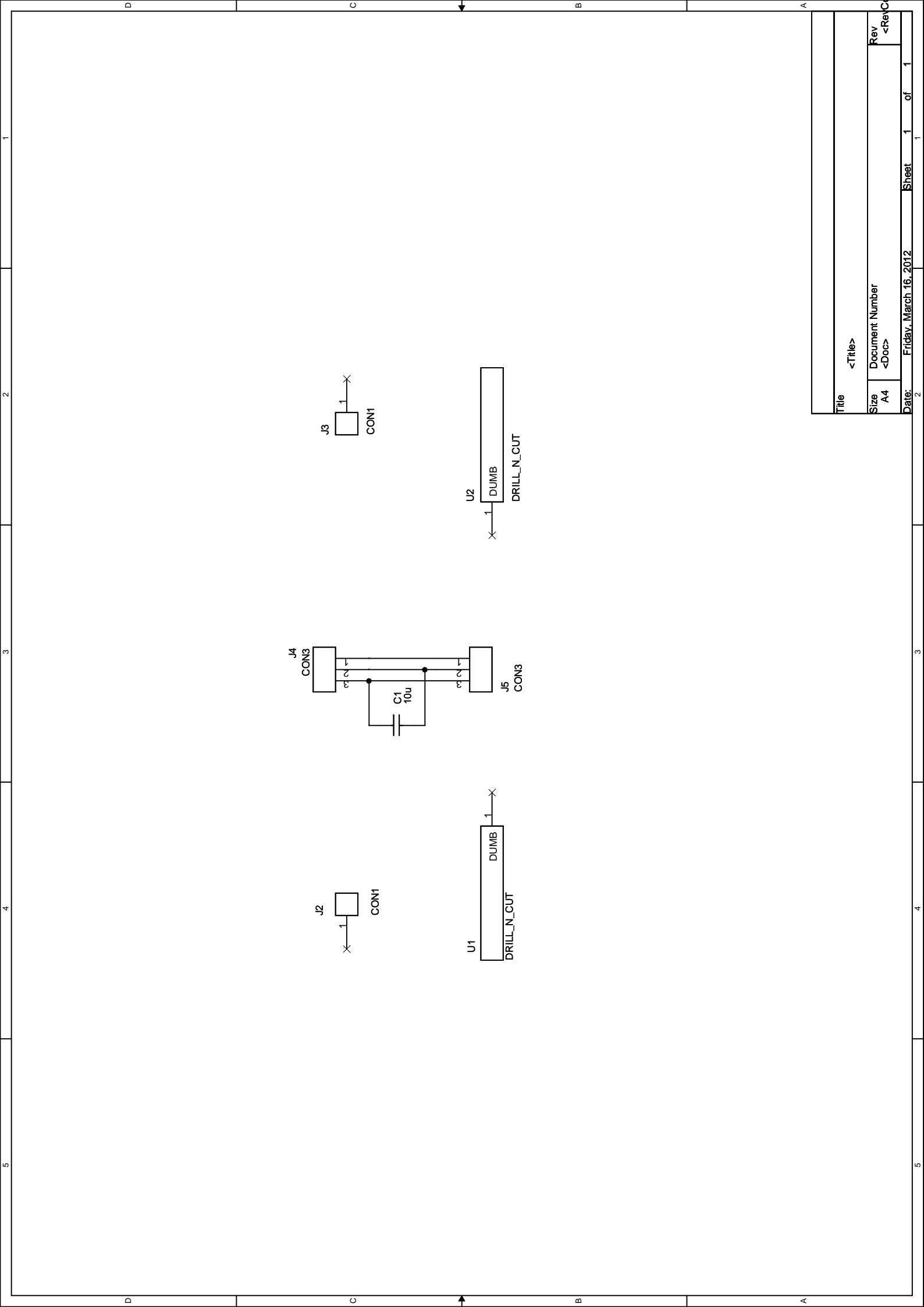


Anexo 3

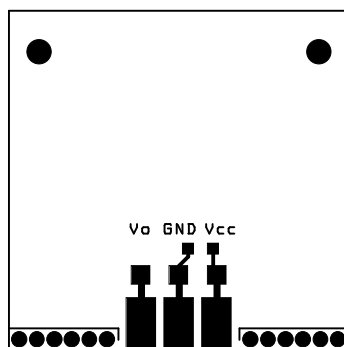
Esquemático e *layout* da PCB dos sensores de Obstáculos

Neste anexo é apresentado o esquemático e *layout* da PCB que serve para acoplar os sensores de obstáculos à PCB que constitui a base da plataforma robótica.

Nas próxima página é mostrado o esquemático e, na seguinte, o *layer TOP* do *Layout* da PCB.



Title		<Title>	
Size	Document Number	Rev	<Rev>
A4	<Doc>		
Date: Friday, March 16, 2012		Sheet	1 of 1



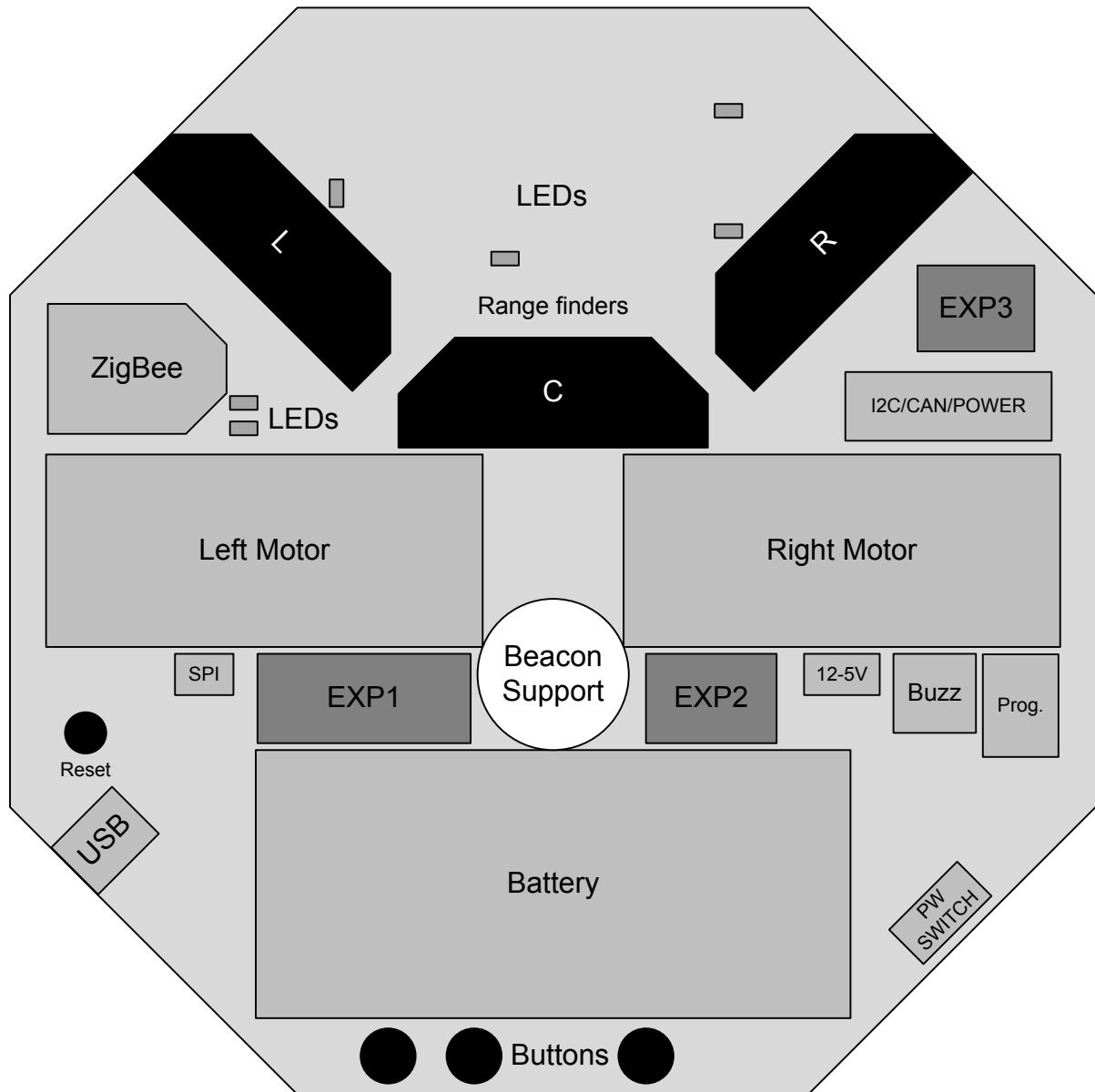
Anexo 4

Esquemas de várias vistas da Plataforma

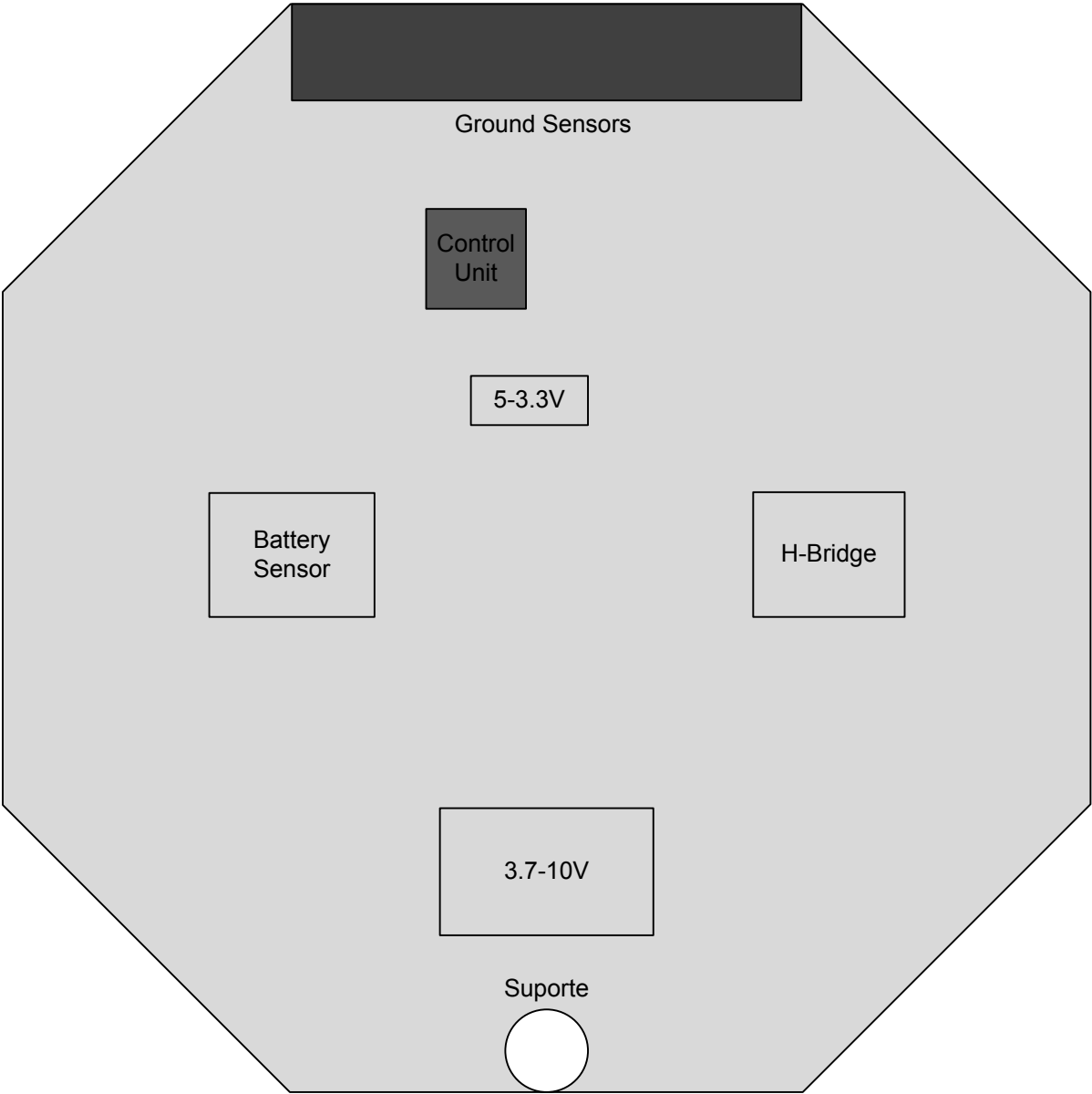
Neste anexo são apresentados um conjunto de esquemas correspondentes a várias vistas da estrutura da plataforma.

Nas páginas 1 e 2 são apresentadas duas vistas, a primeira de cima e a segunda de baixo, da plataforma robótica. Nestas é possível ter uma ideia das posições relativas e absolutas dos vários constituintes desta. Na página 3 pode ver-se um esquema das áreas de deteção dos sensores de obstáculos e as suas posições na plataforma, já na página seguinte pode-se ver a localização dos sensores de superfície bem como um exemplo de uma linha que pode ser usada para o robô seguir. Nas quatro páginas seguintes é possível ver, respetivamente, as posições dos furos para apoiar a expansão da plataforma, os esquemas das áreas de expansão, as posições dos furos para fixação do suporte para a bateria e um pormenor do *footprint* do motor que ilustra a posição dos furos para a fixação deste.

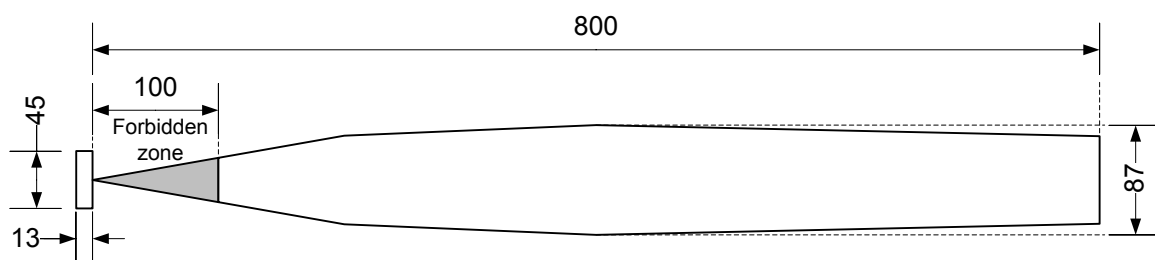
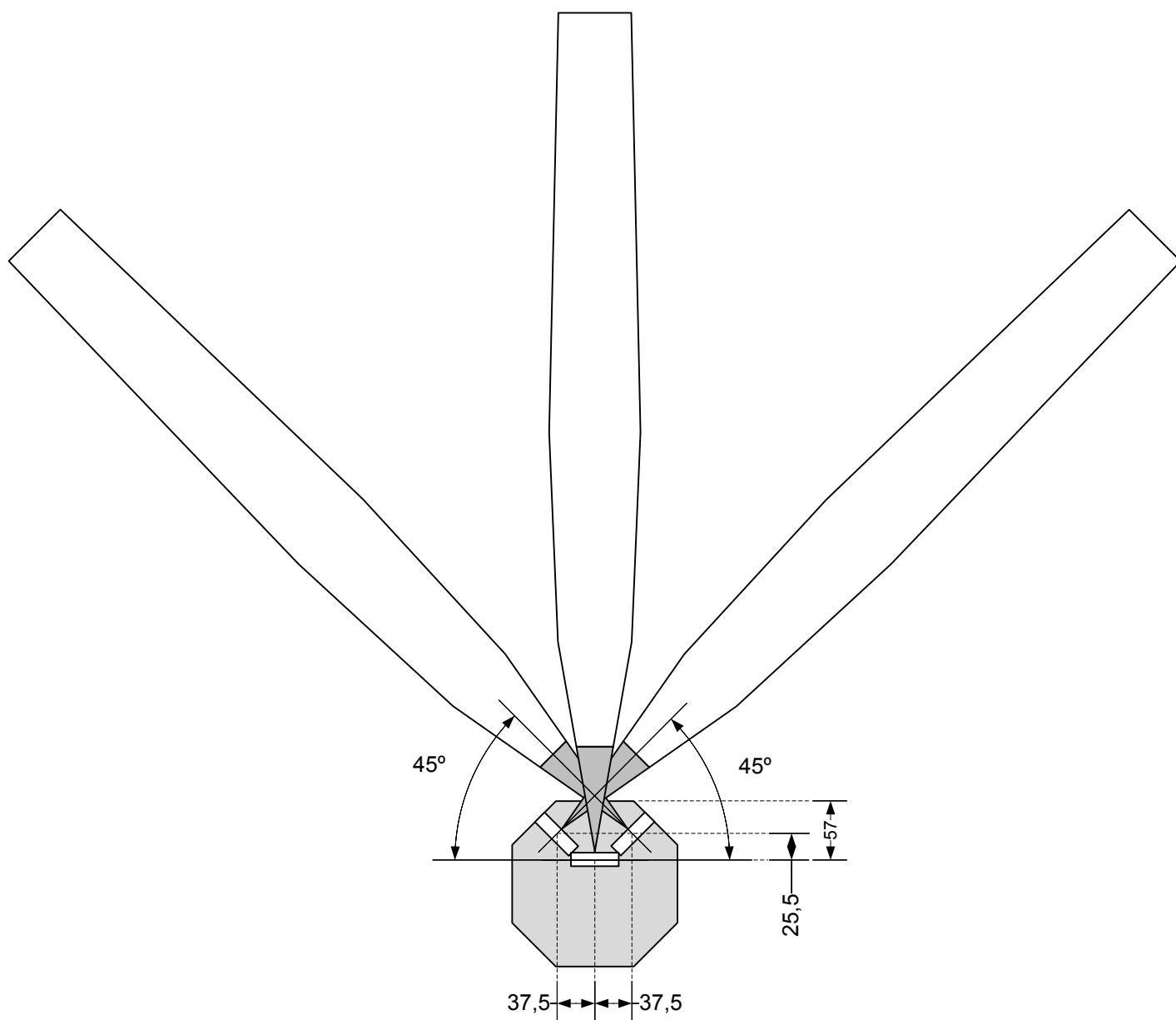
Micro Ratio – Top View



Micro Rato – Bottom View

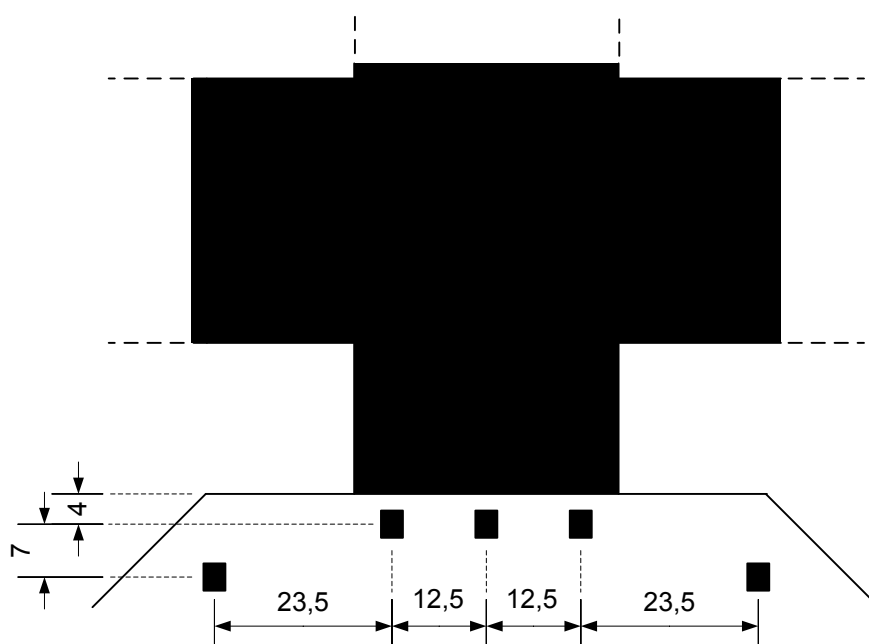


Micro Rato – View Range of Range Finders

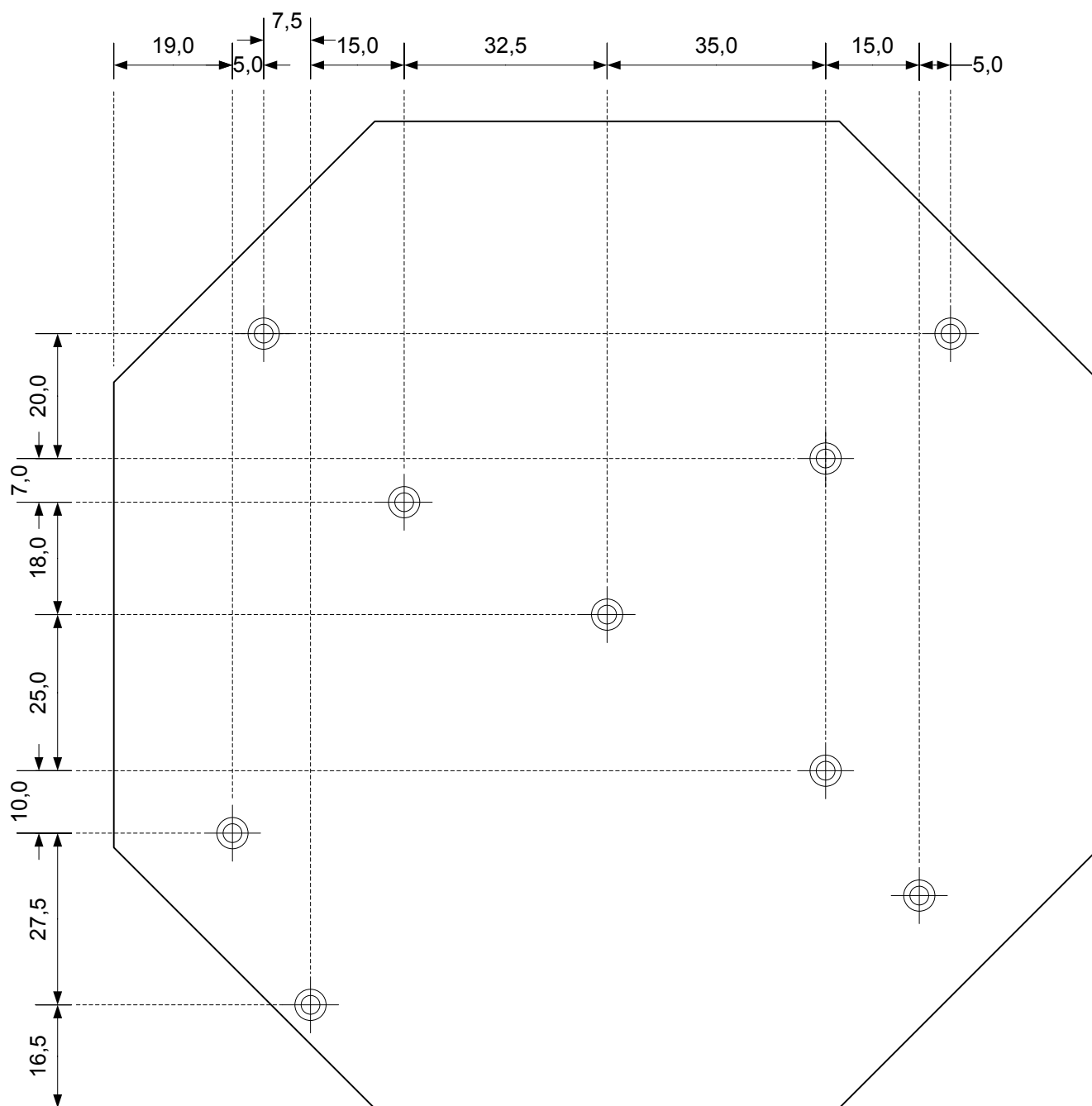


Escala 1:6

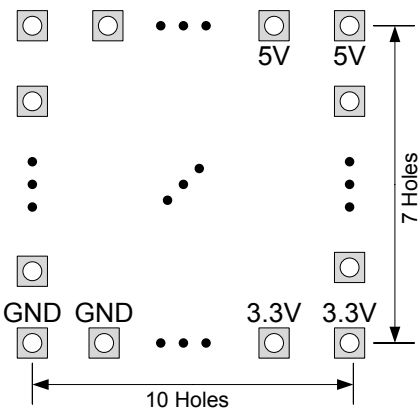
Micro Rato – Ground Sensors Position and Line Example



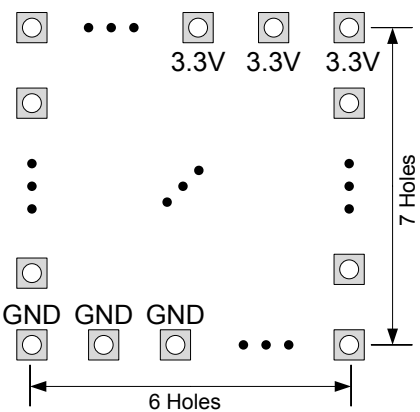
Micro Rato – Position of the Expansion Holes



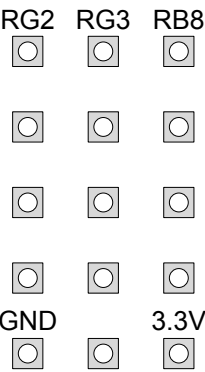
Micro Ratio – Expansion Areas



EXP 1

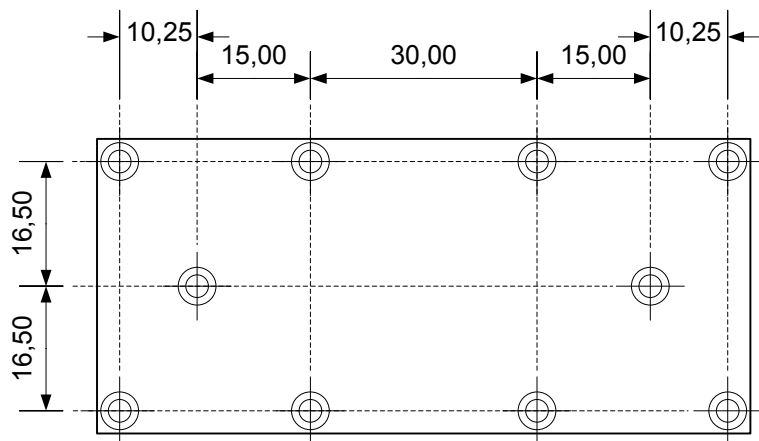


EXP 2

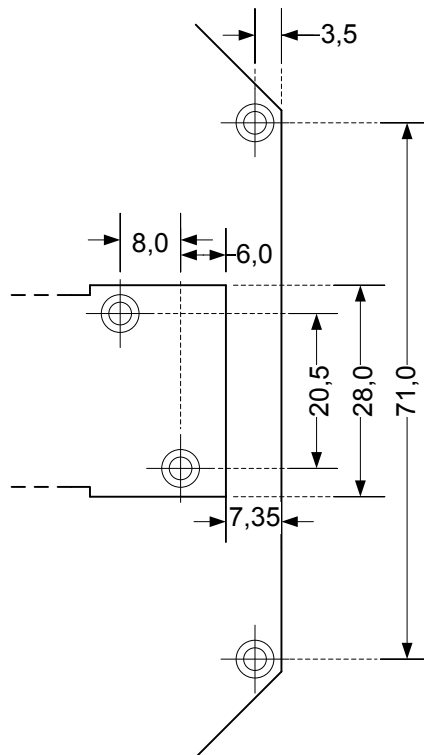


EXP 3

Micro Rato – Position of the Holes in the Battery Footprint



Micro Rato – Motor Footprint Detail



Escala 1:1



3mm Page 8

Anexo 5

Características temporais dos *encoders*

Na Tabela 5.1 são mostradas as características temporais associadas aos *encoders* que foram acoplados aos motores de locomoção. Estas características temporais dizem respeito à Figura 5.1.

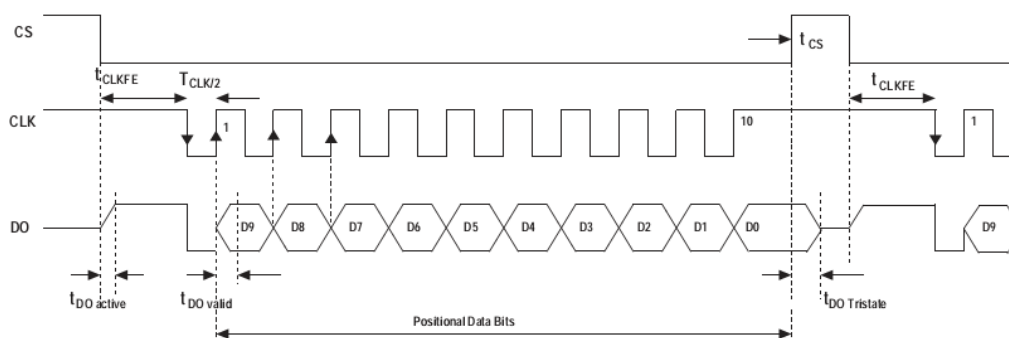


Figura 5.1: Diagrama temporal de uma operação de leitura do *encoder*. [6]

Tabela 5.1: Tempos e frequências associadas ao *encoder* AEAT-6012. Fonte: [6]

Símbolo	Descrição	Valor
$T_{DOactive}$	Tempo entre o <i>falling edge</i> do CSn e ativação da saída de dados	100 ns(MAX)
T_{CLKFE}	Tempo entre o <i>falling edge</i> do CSn e o primeiro <i>falling edge</i> do CLK	500 ns(MIN)
$T_{CLK}/2$	Período mínimo do sinal de <i>clock</i>	500 ns
$T_{DOvalid}$	Tempo entre o <i>rising edge</i> do sinal CLK e saída de dados válida	370 ns(MAX)
$T_{DOTristate}$	Tempo que a DO demora a ficar em alta impedância após <i>rising edge</i> do CSn	100 ns(MAX)
T_{CSn}	Tempo de guarda antes de iniciar nova leitura	500 ns(MIN)
T_{CF}	Tempo de start up da versão de 12 bits	20 ms

Anexo 6

Definição do protocolo de comunicação

Este protocolo define a comunicação entre a plataforma robótica e uma entidade externa para a qual o primeiro envia informação sensorial e espera instruções para execução. São definidas as tramas que circulam entre os intervenientes na comunicação bem como algumas características temporais.

As tramas são compostas por sequências de caracteres *American Standard Code for Information Interchange* (ASCII) para que a sua visualização num terminal série seja mais simples tornando assim o debug mais fácil. Quando se tratam de dados numéricos estes são enviados em base 16 (hexadecimal) sendo representados com caracteres ASCII, assim 1 byte de informação resulta em 2 bytes de dados na trama, uma vez que cada carácter ASCII tem um tamanho de 1 byte

As tramas do protocolo de comunicação têm a estrutura que é apresentada abaixo, sendo que, de seguida, é explicado cada um dos campos da trama.

% CC NN & CS #

- % - Carácter de inicialização de trama (1 carácter);
- CC - Caracteres de comando (2 caracteres);
- NN - *Payload* (tamanho variável, para cada comando espera-se um determinado tamanho);
- & - Carácter de finalização da zona de dados (1 carácter);
- CS - Checksum, soma individual do valor hexadecimal dos caracteres desde o primeiro C até ao último N (2 caracteres);
- # - Carácter de finalização de trama (1 carácter).

Os comandos possíveis são apresentados nas Tabelas 5.2 e 5.4.

Para a comunicação no sentido da plataforma para a entidade externa, cuja única trama a circular foi definida na Tabela 5.2 interessa ainda referir a estrutura desta em mais pormenor para saber o que significa cada campo em específico. Assim, podemos observar na Tabela 5.3 a explicação de cada um dos parâmetros. A comunicação neste sentido não é efetuada até que a entidade externa deliberadamente inicie a comunicação. Uma vez iniciada, a plataforma envia esta trama a cada 40 ms, sendo que a primeira deve ser enviada aproximadamente 80 ms após o início da comunicação. e considera-se que a comunicação foi perdida após um período

Tabela 5.2: Comandos para comunicação da plataforma para a entidade externa.

Comando	Descrição	Tamanho do payload (# characters ASCII)
0x01 - C_SENS	O <i>payload</i> contém a informação atual das leituras de todos os sensores presentes na plataforma	28

Tabela 5.3: Campos de dados da trama de comunicação dos dados sensoriais

Campo	Descrição	Tamanho do campo (# characters ASCII)
Sensores de Obstáculos	Leituras mais recentes dos sensores de obstáculos, do da esquerda para o da direita, valor lido diretamente da ADC com resolução de 8 bits	6
Sensores de Bateria	Leituras mais recentes do sensor de bateria, os três sensores, do da 3 célula para a primeira, valor lido da ADC com resolução de 8 bits	6
Sensores de Superfície	Leituras mais recentes dos sensores de superfície, do sensor 1 para o sensor cinco, o valor enviado é o <i>elapsed time</i> em múltiplos de 50 μ s	10
Odometria	Odometria das rodas de locomoção, atualização desde a última vez que uma trama foi enviada, valor em número de pulsos lidos do <i>encoder</i>	4
Leitura dos Botões	Leitura dos botões da plataforma, informação nos 3 bits menos significativos, pela ordem: <i>start</i> , <i>stop</i> , <i>extra button</i> 1 indica botão premido	1
Sensor de farol	O bit menos significativo indica se o farol se encontra na direção apontada 1 indica farol detectado	1

de 0,5s sem que a entidade externa envie qualquer trama. Após a perda de comunicação a plataforma deixa de enviar dados sensoriais.

Qualquer um dos comandos apresentados na Tabela 5.4 é válido para iniciar a comunicação. No caso de não se pretender atuar sobre a plataforma a trama com o comando C_DUMMY pode ser usada apenas para que a plataforma não declare a cessação da comunicação. Mais uma vez interessa especificar, para as tramas que têm um *payload* maior que 0, o que significa cada campo. A descrição de cada campo é feita na Tabela 5.5 para todas as tramas referidas.

Tabela 5.4: Comandos para comunicação da entidade externa para a plataforma.

Comando	Descrição	Tamanho do payload (# characters ASCII)
0x11 - C_DUMMY	Trama sem <i>payload</i> apenas para manter a comunicação ativa	0
0x12 - C_SPEED	Atuação sobre a velocidade dos motores	4
0x12 - C_LEDS	Atuação sobre os LEDs presentes na plataforma	1
0x12 - C_BUZZER	Actuação sobre o o buzzer	2
0x12 - C_SERVO	Atuação sobre o sevo motor	2

Para completar a Tabela 5.5 é necessário acrescentar alguns pormenores sobre a natureza dos dados que são enviados para atuação da plataforma. No caso dos comandos C_SPEED , C_LEDS e C_SERVO as ordens de atuação recebidas pela plataforma mantêm-se válidas até nova ordem ou até cessação da comunicação. Já no caso do comando C_BUZZER estes são apenas válidos até que o tempo durante o qual o *buzzer* deva apitar se esgote.

Tabela 5.5: Campos de dados da trama de comunicação dos dados sensoriais

Campo	Descrição	Tamanho do campo (# characters ASCII)
Velocidade dos motores, comando C_SPEED	Setpoint para a velocidade dos motores em número de pulsos que este deve percorrer entre cada ciclo de compensação.	4
Estado dos LEDs, comando C_LEDS	Atuação sobre os LEDs da plataforma, cada um dos 3 bits menos significativos define o estado em que os LEDs, do LED 3 para o 1, devem ficar após a execução desta trama, 1 significa LED ligado.	1
Atuação sobre o <i>buzzer</i> , comando C_BUZZER	Tempo durante o qual o <i>buzzer</i> vai apitar, em múltiplos de 80 ms	2
Posição do servo	Posição para a qual o servo motor que serve de apoio ao sensor de farol se deve deslocar. Este valor deve estar num intervalo de -15 a 15 representado valor com igual espaçamento angular no intervalo de rotação do servo-motor	2

Anexo 7

Estimativa de custo da plataforma

Para realizar esta estimativa partiu-se do princípio que o material eletrónico necessário vai ser comprado num dos seguintes fornecedores: *Digikey*, *Mouser*, *RS* ou *Farnell*. Partiu-se de vários pressupostos para ajudar na estimativa que são apresentados na lista a seguir:

- Parte-se do princípio que serão construídas 5 plataformas para que os custos do material comprado em maior quantidade seja mais baixo;
- Sempre que exista uma quantidade mínima de componentes a encomendar e esta exceda os necessários assume-se que se compra essa quantidade e apresenta-se o preço individual desses componentes.
- Os componentes comprados a outros fornecedores diferentes dos referidos serão assinalados e será referido qual é o fornecedor, para os outros não se fará essa distinção.
- Não será considerado o preço das peças de suporte, por exemplo, as peças de suporte aos motores, uma vez que se considera que estas ficarão ao encargo de quem for construir a plataforma. Não será considerado qualquer custo de mão-de-obra uma vez que a montagem da plataforma pode até ser feita por alunos da Universidade de Aveiro.
- Não será considerado o custo da bateria.
- Ter-se-á em conta o custo do sistema como um todo, ou seja, incluindo também o módulo de comunicação construído para o lado da entidade externa e as PCBs de suporte para os sensores de obstáculos.
- Considera-se que apenas a PCB correspondente à plataforma robótica não será construída no DETI

Pode, então, construir-se a Tabela 5.6 onde se apresenta os preços dos componentes eletrónicos necessários para a montagem da plataforma.

A somar ao total da Tabela 5.6 há ainda os custos da construção da PCB que compõe a base da plataforma. Para estimar o custo desta usou-se a ferramenta *online price calculator* disponibilizada pelo fabricante escolhido que foi a Eurocircuits [63]. Esta ferramenta permite calcular o preço de produção de uma PCB tendo em conta o seu tamanho e a quantidade a produzir. Com a introdução destes dados chegou-se ao valor de 39,93 EUR. Assim o custo total de produção do robô, com os pressupostos assumidos é de 294,18 EUR.

Tabela 5.6: Preço do componentes eletrônicos necessários para a montagem do robô

Componentes	Preço(EUR)	Distribuidor
Resistências	2,23	
Condensadores	8,42	
Bobinas	5,01	
Díodos	3,88	
Transístores	4,22	
Conectores	7,82	
Cristal 8MHz	0,30	
Botões/Switches	4,84	
Motores GHM-03 (x2)	32,44	Lynxmotion
Encoders AEAT-6012 (x2)	54,50	
Conversor USB UART FT232RQ	3,71	
Conversor USB UART MM232R	14,85	
Microcontrolador PIC32MX795F512H	8,87	
Opamp MC33184/SO	1,72	
Regulador OKI-78SR	3,63	
Regulador TC1262-3.3	0,52	
Sensores GP2D12 (x3)	27,63	
Sensores QRE1113 (x5)	2,84	
Ponte-H TB6612FNG	2,66	
Transceiver CAN MAX3051	2,61	
Módulos Xbee (x2)	37,34	
Regulador Switching MAX669	6,27	
Servo Motor HS-5055MG	16,74	Servo Shop
Sensor de farol GP1UE28	1,20	
Total	254,25	